Characterizing Electrodermal Responses during Sleep in a 30-day Ambulatory Study

by

Sara Ann Taylor
B.Sc., Brigham Young University, 2014

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in Partial Fulfillment of the Requirements for the Degree of

Masters of Science in Media Arts and Sciences
at the
Massachusetts Institute of Technology

September 2016

© 2016 Massachusetts Institute of Technology. All rights reserved.

Signature of Author __________________________________________

Program in Media Arts and Sciences
August 5, 2016

Certified by _________________________________________________

Rosalind W. Picard
Professor of Media Arts and Sciences
Thesis Supervisor

Accepted by _________________________________________________

Pattie Maes
Academic Head, Program in Media Arts and Sciences
Characterizing Electrodermal Responses during Sleep in a 30-day Ambulatory Study

by

Sara Ann Taylor
B.Sc., Brigham Young University, 2014

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on August 5, 2016
in Partial Fulfillment of the Requirements for the Degree of
Masters of Science in Media Arts and Sciences
at the
Massachusetts Institute of Technology

ABSTRACT

Electrodermal activity (EDA) refers to the electrical activity measured on and under the surface of the skin and has been used to study sleep, stress, and mood. While gathering this signal was once confined to the laboratory, it can now be acquired in ambulatory studies through commercially available wearable sensors. In this thesis, we model and analyze electrodermal response (EDR) events (1-5 second peaks in the EDA signal) during sleep in an ambulatory study.

In particular, we describe an EDR event detection algorithm and extract shape features from these events to discuss the difference in shape between sleep and wake. We also describe an automatic artifact detection algorithm that we use on over 100,000 hours of EDA data we have collected in the 30-day SNAPSHOT Study from 164 participants. Finally, we model the detected EDR events as a point process using a state-space generalized linear model. We identify a significant influence of recent EDR event history on current EDR event likelihood across different participants. We also use this model to analyze EDR event rates during different periods of the night.

Thesis Supervisor: Rosalind W. Picard
Title: Professor of Media Arts and Sciences
Characterizing Electrodermal Responses during Sleep in a 30-day Ambulatory Study

by

Sara Ann Taylor
B.Sc., Brigham Young University, 2014

Advisor
Dr. Rosalind W. Picard
Professor of Media Arts and Sciences
MIT Media Lab

Reader
Dr. Emery N. Brown
Edward Hood Taplin Professor of Medical Engineering
Professor of Computational Neuroscience
Warren M. Zapol Professor of Anaesthesia
Massachusetts Institute of Technology

Reader
Dr. Pattie Maes
Professor of Media Arts and Sciences
MIT Media Lab
# Contents

Abstract 3  
List of Figures 9  
List of Tables 10  
Acknowledgements 11  

1 Introduction 13  

2 SNAPSHOT Study 15  
2.1 General Background  . 15  
2.2 Data Sources 17  
  2.2.1 Standardized Surveys 17  
  2.2.2 Daily Surveys 17  
  2.2.3 Hospital Overnight 21  
  2.2.4 Phone Monitoring App 22  
  2.2.5 Wearables 22  

3 Electrodermal Activity Data 27  
3.1 Background 27  
3.2 Electrodermal Response (EDR) Event Detection 28  
3.3 Electrodermal Response Features 29  

4 Artifact Detection 33  
4.1 Introduction 33  
4.2 Automatic Identification of Artifacts in Electrodermal Activity Data 34  
4.3 Active Learning for Electrodermal Activity Classification 35  

5 Point Process Modeling 37
5.1 Background ........................................ 37
5.2 State-Space Generalized Linear Model ..................... 38
  5.2.1 Observation Equation .................................. 38
  5.2.2 State Equation ........................................ 40
  5.2.3 Estimation of Model Parameters using Expectation Maximization 41
  5.2.4 Goodness-of-Fit ....................................... 41
  5.2.5 Model Selection ....................................... 42
  5.2.6 Special Cases of the State-Space Generalized Linear Model ...... 42
5.3 Modeling EDR events during sleep .......................... 43
5.4 Results .................................................. 49
  5.4.1 Selected Models ....................................... 49
  5.4.2 Model Fit on Held-out Data ............................. 51
  5.4.3 History Effects ....................................... 54
  5.4.4 Time Since Sleep Onset Results ......................... 54
5.5 Summary ................................................. 57
6 Conclusions and Future Work ................................. 61
  6.1 Summary of Work ....................................... 61
  6.2 Implications ............................................ 62
  6.3 Future Work ............................................ 63
A Automatic Identification of Artifacts in Electrodermal Activity Data 65
B Active Learning for Electrodermal Activity Classification 71
C State-Space Generalized Linear Model Derivation .......... 79
  C.1 Model Set-up ........................................... 79
    C.1.1 Definitions of Variables ............................... 79
    C.1.2 Observation Equation - Conditional Intensity Function (CIF) .... 80
    C.1.3 State Equation ...................................... 81
  C.2 Expectation-Maximization ................................ 81
    C.2.1 Complete Data Likelihood ............................. 81
    C.2.2 E-Step .............................................. 82
    C.2.3 M-Step ............................................. 90
    C.2.4 EM Algorithm Summary ................................ 96
Bibliography .................................................... 97
List of Figures

Figure 2.1 Number of Participants in Each Cohort ................................................. 16
Figure 2.2 Number of Participants in Each Year in School ................................. 16
Figure 2.3 Distribution of Stress and Anxiety Measures ...................................... 18
Figure 2.4 Distribution of Pre-study MCS Scores and Change in MCS Scores .... 19
Figure 2.5 Distribution of Daily Self-reported Wellbeing Measures .................... 20
Figure 2.6 Distribution of Self-Reported Daily Behaviors ................................... 21
Figure 2.7 Distribution of DLMO ........................................................................... 22
Figure 2.8 Distribution of Average Time Spent with Phone Screen On per Hour .... 23
Figure 2.9 Distribution of Self-Reported Sleep Accuracy ..................................... 24
Figure 2.10 Distribution of Sleep Regularity Scores .............................................. 24
Figure 2.11 Example of Q-Sensor Data ................................................................. 25
Figure 3.1 Typical EDR ....................................................................................... 28
Figure 3.2 Example of Identified EDRs ................................................................. 29
Figure 3.3 Labeled EDR Features ......................................................................... 30
Figure 3.4 Comparison of Sleep and Wake EDR Features .................................... 31
Figure 5.1 Example of Detected Artifacts During Sleep ....................................... 44
Figure 5.2 Example of Detected Artifacts During Wake ...................................... 44
Figure 5.3 EDR Sleep Raster Plot ......................................................................... 45
Figure 5.4 Histogram of Inter-event Interval (IEI) Times ..................................... 46
Figure 5.5 Distribution of Median of Percentage of Artifact-free Data ................ 46
Figure 5.6 Distribution of Number of Sleep Episodes with No Detected Peaks .... 47
Figure 5.7 Held-out K-S Statistic for Static-GLM models ..................................... 52
Figure 5.8 Held-out K-S Statistic for SS-GLM models .......................................... 53
Figure 5.9 Connection between Sleep Regularity and K-S Statistic in SS-GLM .... 54
Figure 5.10 History Parameters ............................................................................ 55
Figure 5.11 Time Since Sleep Onset Effect for Two Participants ......................... 59
Figure 5.12 Example of Rate Differences and Self-reported Health .................... 60
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Computed EDR Feature Values</td>
<td>30</td>
</tr>
<tr>
<td>5.1</td>
<td>Selected PSTH Models using Minimum AIC Score</td>
<td>49</td>
</tr>
<tr>
<td>5.2</td>
<td>Selected Static GLM Models using Minimum AIC Score</td>
<td>50</td>
</tr>
<tr>
<td>5.3</td>
<td>Selected SS-GLM Models using Minimum AIC Score</td>
<td>50</td>
</tr>
<tr>
<td>5.4</td>
<td>Comparing Models using AIC and Goodness-of-Fit Metrics</td>
<td>51</td>
</tr>
<tr>
<td>5.5</td>
<td>Model Ranking based on AIC and K-S Statistic</td>
<td>51</td>
</tr>
<tr>
<td>5.6</td>
<td>Final Selected SS-GLM Models to be Analyzed for Inference</td>
<td>55</td>
</tr>
<tr>
<td>5.7</td>
<td>Empirical Bayes Estimates of EDR Rates during Different Periods</td>
<td>57</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

I would first like to thank Roz for her willingness to give me space to explore and for encouraging me to solve hard problems. I would also like to thank my readers, Emery Brown and Pattie Maes for their helpful feedback.

To the Affective Computing Group (Asma, Natasha, Vincent, Daniel, Ehi, Kristy, Craig, Grace, Karthik, Akane, Javier, Szymon, Yadid, and Micah): thank you for helping me improve my research and encouraging me to work on exciting problems.

Extra special thanks is due to Akane Sano for allowing me to collaborate with her on the SNAPSHOT study. Additionally, I express my gratitude to Natasha Jaques for pushing us to work on exciting projects and to Yadid Ayzenberg for help in using Tributary to speed up the analysis.

I have also had the pleasure to work with many UROP students (Ian, Marisa, Victoria, Yuna, Tim, Sienna, Sarah, Lars, Noah, Kevin, Hank, William, Margaret, and Alicia), whose contributions have not gone unnoticed. And to the Brigham and Women’s Hospital, Harvard, and the Macro Connection teams, the SNAPSHOT Study could not have been conducted without your help: thank you!

A big thank you goes to my family, and especially my parents, for their love, support, and encouragement.

And finally, to my best friend and husband, Cameron: thank you for your loving support and willingness to go on this adventure with me.
Chapter 1

Introduction

Many clinical studies have shown how sleep affects us physiologically, mentally, and emotionally, and how different behaviors influence sleep; however, these studies are usually conducted in sleep labs with many devices and sensors that inhibit natural sleeping behaviors. Additionally, these studies are usually conducted in carefully controlled environments. While these studies have helped us begin to understand the role of sleep in humans, they are not directly applicable to everyday life without further research because of the limits that a sleep lab places on natural sleep and behaviors. However, with the rise of smartphones and wearable physiological sensors, researchers now have the tools to objectively measure sleep and behaviors during daily life.

One such physiological measure that can be used to monitor both sleep and wake is electrodermal activity. Electrodermal activity (EDA) refers to the electrical activity measured on and under the surface of the skin [7]. EDA has frequently been used in studies related to affective phenomena and stress because EDA tends to increase as the body responds to arousal or exertion (e.g., [20, 21, 26, 27, 34, 36, 44]). In addition, increases in EDA have also been observed during sleep when arousal and exertion are typically low. In particular, it has been shown that if “sleep storms” (the pattern of many successive electrodermal responses during sleep) are present they usually occur during slow-wave sleep [25, 28].

In this thesis, we will model electrodermal response (EDR) events (peaks in the EDA data that are short in duration) in both their shape and their rate patterns during sleep at home. In particular, we will describe an EDR event detection algorithm, how we extract shape features from each event, and how we analyze the shape feature differences during wake and sleep. In order to do so, we also needed to develop an automatic artifact detection algorithm. Additionally, we describe how EDR events can be represented as a temporal
point process, a random process composed of a series of binary events in a time-series \cite{14}. We use the point process model described by Czanner et. al. \cite{13} to estimate the rate of EDR events during sleep and use the result to model the relationship between self-reported wellbeing and EDR event rates during sleep. In order to model these patterns and relationships, we use data collected from the ongoing SNAPSHOT Study.

Outline

Chapter 2: SNAPSHOT Study gives details on the study and presents information about the population. The remainder of this thesis is based on data collected during this study.

Chapter 3: Electrodermal Activity Data gives background information on electrodermal activity and describes a new technique for identifying electrodermal responses (EDRs). This technique is used to extract shape features for EDRs throughout the day, and sleep and wake EDR shape features are compared.

Chapter 4: Artifact Detection describes the importance of automatically detecting artifacts in EDA data and describes our machine learning methods to do so.

Chapter 5: Point Process Modeling describes a state-space based generalized linear model that can be used to estimate the rate of a point process such as EDR events during sleep. We then describe how the model is fit to EDA data during sleep for each SNAPSHOT participant. From this model, we draw conclusions about the effect of recent history and time since sleep onset on the rate of EDR events during sleep.

Chapter 6: Conclusions and Future Work summarizes the thesis and implications of the results. We also touch on future work to improve upon the methods presented here.
Chapter 2

SNAPSHOT Study

2.1 General Background

The SNAPSHOT Study is a large-scale and long-term study[^1] that seeks to measure: Sleep, Networks, Affect, Performance, Stress, and Health using Objective Techniques. This study investigates:

1. how daily behaviors influence sleep, stress, mood, and other wellbeing-related factors
2. how accurately we can recognize/predict stress, mood and wellbeing
3. how interactions in a social network influence sleep behaviors

To date, approximately 250 undergraduate students have participated in the 30-day observational study that collects data from a variety of sources [35]. These students were recruited as socially connected groups with about 40-50 students participating during each of the 6 cohorts since 2013[^2]. This study is an NIH-funded collaborative research project[^3] between the Affective Computing group at the MIT Media Lab and Brigham and Women’s Hospital. The Massachusetts Institute of Technology Committee On the Use of Humans as Experimental Subjects (COUHES) approved this study and all participants gave informed consent.

In this thesis, we will present results based on the first 4 cohorts only which had a total of 164 participants since the latter two cohorts were conducted as the analysis was being run (see Figure 2.1 for the number of participants that participated in at least some aspect of

[^1]: More details and data visualizations can be found at [http://snapshot.media.mit.edu/](http://snapshot.media.mit.edu/)
[^2]: Note: The first cohort only contained 20 students
[^3]: NIH Grant R01GM105018
the 30-day SNAPSHOT Study). Participants were all undergraduates, but were distributed across all four years of school with freshman being the highest represented group (see Figure 2.2). Our population was overwhelmingly male (104 participants).

![Figure 2.1: Number of Participants in Each Cohort. Each participant only participated during one cohort. Fall cohorts were conducted in October and November and spring cohorts were conducted in February and March.](image1)

![Figure 2.2: Number of Participants in Each Year in School](image2)
2.2 Data Sources

2.2.1 Standardized Surveys

After being informed of the study protocol and providing consent, participants completed several standardized surveys, including Pittsburg Sleep Quality Index test, Myers Brigg Personality test, Big Five Inventory Personality Test, Morningness-Eveningness questionnaire, Perceived Stress Scale, SF-12 survey, and a demographics survey that collected information on age, gender, academic year, academic major, and living situation.

At the end of the 30-day study, participants completed several other post-study surveys, including academic performance during the semester (GPA), Perceived Stress Scale (PSS), SF-12 survey (which gives Physical Component Summary (PCS) and Mental Component Summary (MCS) scores), and the State-Trait Anxiety Index (STAI). See Figure 2.3 for distributions of three of the standardized surveys about stress and anxiety.

We are particularly interested in the Mental Component Summary (MCS) score because of the importance of mental health in overall wellbeing. Figure 2.4 shows the distribution of pre-study MCS scores and the change in MCS scores over the 30-day study (computed as $\Delta = \text{post MCS} - \text{pre MCS}$).

2.2.2 Daily Surveys

Each morning (at 8am) and each evening (at 8pm) during the study, participants were emailed a link to an online survey that they were instructed to complete right after waking and right before retiring to bed, respectively. Each participant’s survey response was looked over by a member of the research staff to catch any errors or missing entries. Participants were given 36 hours to start the survey and 72 hours to correct any mistakes after the survey was initially emailed.

The morning survey included questions about the participant’s sleep, including the time the participant tried to go to bed, sleep latency, wake time, the timing and duration of each awakening or arousal during sleep, and the timing and duration of any naps. Finally, participants rated how they were feeling by adjusting sliders between the following pairs of words: Sleepy — Alert; Sad — Happy; Sluggish — Energetic; Sick — Healthy; and Stressed Out — Calm Relaxed. The slider responses were reported on a scale from 0–100, but participants could not see the numerical value when reporting. Each participant also reported how they were feeling during the evening survey using the same five sliders (see Figure 2.5 for distributions of the responses for each of the five sliders).
We note that only the alertness and energy self-reports appear to have differences between morning and evening surveys; that is, participants self-reported being significantly less alert and less energetic in the evening compared to the morning with \( p < 0.05 \), but no significant difference was found for the other three self-report metrics.

The evening survey included questions about the timing and duration of the participant’s activities, including academic, exercise, and extracurricular activities. The participant was
Figure 2.4: Distribution of (a) Pre-Study MCS Scores and (b) Change in MCS Scores. Higher MCS scores indicate better mental health and positive changes in MCS score indicate improved mental health over the 30-day study.

also asked to report on the number of hours he or she studied alone. Participants reported on caffeine and alcohol intake, consumption of sleeping pills, and use of other medications that might make the participant more alert or more tired (see Figure 2.6 for distributions of different self-reported daily behaviors).
Figure 2.5: Distribution of Daily Self-Reported Wellbeing Measures for the morning and evening surveys.
Figure 2.6: Distribution of the number of days that participants self-reported (a) alcohol consumption, (b) caffeine consumption, and (c) exercise. For example, 5 participants self-reported 25 days of exercise activities, but 22 participants self-reported 0 days of exercise. We note that some participants had more than 30 days of an activity because they were asked to extend their 30-day study due in order to schedule their hospital overnight during the study.

2.2.3 Hospital Overnight

During the course of the 30-day study, each participant was invited to participate in one overnight stay at the hospital. While at the hospital, participants gave saliva samples every hour and completed a test to measure their alertness. From these tests, we were able to compute Dim Light Melatonin Onset (DLMO) for each participant (see Figure 2.7).
2.2.4 Phone Monitoring App

At the beginning of the study, participants downloaded onto their Android Phones a phone monitoring app that collected data, encrypted it, and sent it to our secure servers. This was done passively throughout the study. The app was developed by Akane Sano and was based on the Funf framework[17]. It collected, metadata from SMS and calls including the timestamp, the to/from phone numbers, and the duration of calls. Additionally, the app also collected location data (including location source), app usage data, and screen on/off timing (see Figure 2.8 for distribution of the average time spent with the screen on for each hour of the day).

2.2.5 Wearables

During the 30-day study, participants also wore two wearable sensors: the Actiwatch and Affectiva Q-sensor. Participants were instructed to wear these sensors at all times, except when they might get wet (e.g., during showering or swimming).
The Actiwatch was worn on the non-dominant hand and recorded light intensity levels and acceleration data. These data were aggregated to one-minute sample rates on the wearable itself and, using proprietary software, automatically determined sleep and wake episodes on the scale of 1 minute. These binary labels were then compared to the self-reported sleeping episodes (see Section 2.2.2) by an expert in order to determine when the participants were actually sleeping. We found that participants were, on average, 93.5% accurate in reporting sleep episodes (see Figure 2.9). This means that the average participant reported a different sleep/wake state for approximately 28 hours during the study than the expert determined using the Actiwatch data.

Though there are many measures of sleep regularity, we computed each participant’s sleep regularity score based on the expert-labeled sleep/wake episodes on a scale from 0-100 based on the consistency of sleep/wake patterns on a 24-hour period (with sleep regularity score of 100 being perfectly regular, e.g., same bedtime and wake time each day). In order to compute the sleep regularity score with a period of 24 hours, we let \( s(t) \) be the indicator function with \( s(t) = 1 \) during wake and \( s(t) = -1 \) during sleep, \( \tau = 24 \) hours and \( T \) = total number of hours of data, then

\[
\text{Sleep Regularity Score} = 200 \cdot \left( \frac{\int_{0}^{T-\tau} s(t)s(t+\tau)dt}{T - \tau} - 0.5 \right)
\]  

(2.1)
Figure 2.9: Distribution of how accurate participants were in self-reporting sleeping episodes compared to the expert label.

This score was defined and computed by the Brigham and Women’s Hospital (BWH) Team of the SNAPSHOT study and has been used in several of their publications [4, 11, 12]. The distribution of sleep regularity scores for our population is given in Figure 2.10.

Figure 2.10: Distribution of Sleep-Regularity Scores. See the Equation 2.1 for how the sleep regularity score was computed.
Q-Sensor

The Q-Sensor was worn on the dominant hand and recorded electrodermal activity (EDA), skin temperature, and 3-axis accelerometer data, at a sample rate of 8Hz. An example of a 9 hour period of data from the Q-Sensor is given in Figure 2.11. We focus on electrodermal activity signals for much of the remaining chapters of this thesis.

Figure 2.11: Example of Q-Sensor Data from one participant over a 9 hour period (from 2am to 11am). We note that the participant was sleeping from 2am to 7:40am which can easily be seen in the acceleration data.
Chapter 3

Electrodermal Activity Data

3.1 Background

Electrodermal activity (EDA) refers to the electrical activity measured on and under the surface of the skin [7]. EDA has frequently been used in studies related to affective phenomena and stress because EDA tends to increase as the body responds to arousal or exertion (e.g., [20, 21, 26, 27, 34, 36, 44]). In addition, increases in EDA have also been observed during sleep when arousal and exertion are typically low. In particular, it has been shown that if “sleep storms” (the pattern of many successive EDA responses during sleep) are present they usually occur during slow-wave sleep [25, 28].

In general, electrodermal activity analysis splits the signal into tonic and phasic signals. Tonic signals refer to the slow-moving level of electrodermal activity, whereas phasic signals, also called electrodermal responses (EDRs), are short in duration and have a characteristic shape. Boucsein provides a description of the characteristic shape of an EDR: the response typically lasts between 1-5 seconds [1], has a steep onset and an exponential decay, and reaches an amplitude of at least 0.01 µS (see Fig. 3.1 for an example of a typical EDR) [7]. These responses were originally studied in research labs and were observed after a stimulus was presented to the study participant. However, EDRs have also been observed when no stimulus appeared to be present; these are often called spontaneous EDRs.

In the following sections, we will discuss different methods of detecting EDRs and the

---

1 There are several different methods of measuring electrodermal activity, including measuring skin conductance. If skin conductance is measured, electrodermal responses are also called skin conductance responses (SCRs).

2 While a single EDR typically lasts 1-5 seconds, EDRs can overlap; that is, another EDR can begin before the first has finished.
Figure 3.1: A typical EDR selected from a SNAPSHOT participant’s data. We can see that this EDR lasts about 5 seconds, and it has the typical steep onset and exponential decay.

different features that can be computed from them. We note that because we have collection trillions of data points over the course of the study, we needed an efficient method to process the data. We used Tributary, an interactive distributed analytics platform for large-scale time series data [1], to simply and efficiently process the EDA data.

3.2 Electrodermal Response (EDR) Event Detection

There exist several methods for identifying EDR events in EDA data. One of the most popular methods is the Ledalab package for MATLAB [5, 6]. This package uses convolution methods to identify EDRs by assuming that the shape of the EDRs for a single person remains the same across time. A newer method called cvxEDA uses convex optimization to split the data into phasic, tonic, and noise components to identify EDR events [18]. This method also assumes that the shape of the EDRs is stable for the same person. While both of these methods have their merits, we have noticed that the EDR structure may change for a person, thus violating the assumptions of both techniques. Furthermore, both methods appear to have been designed for laboratory settings, while we are interested in ambulatory observations where a more diverse experience may elicit different EDRs.

Therefore, we use a simple peak detection algorithm in order to find EDRs. In particular, we identify the apex of the EDR as a peak in the EDA data (after it has been low-pass filtered with an FIR-filter with 64 taps and 1Hz cutoff) that has a derivative of at least $0.064 \mu S/s$ within 0.5 seconds prior to the identified apex (see Figure 3.2). We note that these thresholds are similar to what has been used in the past (e.g., $0.01 \mu S/s$ and $0.09 \mu S/s$
were used by Sano et. al. [38, 37], and 0.093μS/s was used by Healey [19]. This method tends to over-estimate the number of peaks identified compared to Ledalab and cvxEDA methods, but we have created an artifact detection algorithm to reduce the number of false positives (see Chapter 4 for more details).

![EDA](image)

Figure 3.2: Example of identified EDRs (apex identified by vertical red lines) detected using our peak detection algorithm on a SNAPSHOT participant’s data. Less abrupt peaks at approximately 5 and 50 seconds are not included with this particular set of parameters.

### 3.3 Electrodermal Response Features

Once an EDR has been detected, we can compute several features about the basic shape as suggested by Boucsein [7]: apex value, rise time, decay time, width, amplitude, and maximum derivative. In order to compute these features, we also need to compute the start and end times of the EDR. Boucsein suggests that the start of the EDR is when the derivative drops below 1% of its maximum value and the end time to be defined as the time when the signal drops to less than 50% of the amplitude of the EDR. The width of the EDR is defined as the time delta between when the signal reaches 50% of the amplitude during the rising phase to when the signal falls below 50% of the amplitude on the decaying side.

---

3All of these thresholds were originally chosen to conform with the theory of EDRs and were found to be appropriate based on visual inspection; however, future research into better detection algorithms that don’t assume a single EDR response is needed. For example, a multi-threshold approach might work well.
Figure 3.3 displays the shape features on an EDR and Table 3.1 gives the computed values of the example EDR.

![Figure 3.3: After a peak has been detected, we can compute a number of features, including amplitude, rise time, width, and AUC (approximated by multiplying the width by the amplitude [7]). See the text for details on computation and Table 3.1 for computed values.](image)

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Apex Value</th>
<th>Rise Time</th>
<th>Width</th>
<th>Amplitude</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>5.055 µS</td>
<td>2.125 s</td>
<td>2.375 s</td>
<td>0.385 µS</td>
<td>0.915 µS · s</td>
</tr>
</tbody>
</table>

Additionally, we can plot histograms of these shape parameters to see how the shape of EDRs for one person can vary within and between sleep and wake (see Figure 3.4). In particular, we note that the apex value and rise times of identified EDRs during sleep were much larger than those identified during wake for this participant on a single day.

In addition to features computed on a single EDR, researchers are often interested in the frequency or rate at which these responses appear. Because these responses appear with irregular patterns, Fourier analysis has not been found to be useful. Generally, simple features (e.g., the number of events in a time window) are computed in order to approximate the rate of EDR events. However, these simple methods of feature extraction are lacking in the ability to capture both the historical effects (when an EDR is detected in the recent history, how does that effect the current probability of an EDR event) and the time-of-day or day-in-sequence effects. We will discuss a better way to capture rate features in Chapter 5.
Figure 3.4: Each subfigure shows a different EDR feature during sleep (blue) and wake (red) for one day’s worth of data for a single SNAPSHOT Participant. There were 456 EDRs identified during 5 hours and 40 minutes of sleep and 558 EDRs identified during 16 hours and 20 minutes of wake.
Chapter 4

Artifact Detection

4.1 Introduction

As discussed in Chapter 3, electrodermal activity (EDA) has been used in several studies in order to relate affective phenomena and stress (e.g., \[20, 21, 26, 27, 34, 36, 44\]). As these studies have moved from the laboratory environment to ambulatory studies, there is a much greater need to detect artifacts, or noise, in the EDA signal. Electrodermal artifacts typically arise from movement of the skin or muscles beneath the electrodes or pressure changes causing a change in the surface area of the skin in contact with the electrodes \[7\]. These artifacts are especially likely during ambulatory studies with wrist-worn sensors (like SNAPSHOT) where the participant is going about his or her normal activities which may include walking, riding in a vehicle, typing on a keyboard, or writing.

The most typical way to remove artifacts is by low-pass filtering the EDA signal or using exponential smoothing \[21, 27, 33, 36\]. This technique is able to remove small noise variations, but is unable to handle the large increases or decreases in EDA signal that often occur when the pressure on the electrodes changes. Others have used heuristic techniques to identify artifacts by looking for abnormal signal variations (e.g., very rapid increases or decreases, or small EDR widths) \[20, 27, 41\]; however, these techniques only have established thresholds for particular studies, and were only verified through visual inspection. Therefore, these heuristic measures are not guaranteed to generalized to other settings. Others have suggested only considering data during which there is no movement (as determined by an accelerometer). This method, however, cannot detect artifacts due to pressure changes or skin-to-electrode contact changes. The final, least sophisticated way of identifying artifacts requires an expert to look through the signal and manually identify
portions of the signal containing artifacts.

Because we have collected over one-hundred thousand hours of EDA data, having an expert label all the data is not feasible. Furthermore, we want to be able to know which portions of our signal contain artifacts without having to rely on heuristic techniques. Therefore, we have created a simple machine learning algorithm to identify which 5-second portions of and EDA signal contain artifacts. We have published two papers on this topic and they are reproduced in their entirety in Appendix A and B. In the following sections, we will give the high-level overview of each paper.

4.2 Automatic Identification of Artifacts in Electrodermal Activity Data

In our first paper published at the IEEE Engineering in Medicine and Biology Conference in 2015\(^1\), we were able to train a simple binary classifier to identify artifacts in five-second portions of EDA data and achieved 95.7% accuracy on a hold-out test set.

We had two experts label 1,301 5-second portions of EDA signal as containing an artifact or clean based on a common criteria\(^2\). We then considered two types of classifiers: binary and multiclass. The binary classifier only used portions of the signal that the experts agreed on, whereas the multiclass classifier included a third class (which we named “questionable”). We then extracted several features from the EDA signal for each 5-second period related to the EDA signal’s shape. The data were then split into 3 data sets: training, validation, and testing. Using Wrapper feature selection, a subset of features was chosen using a training set. After trying several different machine learning methods, support vector machines (SVMs) were chosen to be the best and a sweep of SVM parameters was conducted in order to choose the best SVM model using training and validation sets. Finally, the selected models (one for the binary classifier and one for the multiclass) were tested on the held-out test set to produce the final results.

Furthermore, in order to achieve this result and allow the model to be able to be easily used by other researchers, we developed an online tool that helps labeling portions of uploaded data, identifying EDRs, and automatically detecting artifacts. This tool is called EDA–Explorer and can be found at [http://eda-explorer.media.mit.edu/](http://eda-explorer.media.mit.edu/). This tool allows a group of researchers to join as a team. The team can then upload files in

\(^1\)The full paper is reproduced in Appendix A

\(^2\)The experts agreed on 80.71% of the 5-second portions (see Appendix A for a complete discussion)
several different formats in order to automatically label artifacts or detect peaks using the online system. We also realized that while our criteria for artifacts would work for similar ambulatory studies, they might not work for other studies like seizure detection where large increases in the EDA signal should be treated as a part of the signal instead of noise. Additionally, we realized that there is great value in being able to have a system for manually labeling EDA data. For these reasons, we also created the ability for teams to manually label their own data. Teams can require a certain number of labels for each 5-second epoch and track their labeling progress.

We have also open-sourced our code for researchers that would like to use our classifiers and EDR detection code offline in batch mode. This code is available at https://github.com/MITMediaLabAffectiveComputing/eda-explorer/

4.3 Active Learning for Electrodermal Activity Classification

In our second paper published at the IEEE Signal Processing in Medicine and Biology Symposium (SPMB) in 2015\(^3\), we wanted to expand on our first paper and reduce the human labeling effort by using active learning techniques. We were able to reduce labeling effort by as much as 84% in our applications. This was accomplished by using a variety of SVM-based active learning techniques that intelligently ask the human for labels on 5-second periods.

In comparing the active learning techniques, we used the same dataset as the paper discussed in Section 4.2 and a new data set of EDRs where each 5-second period was labeled as containing an EDR or not. We found that the Simple Margin technique \(^{10}\) performed very well on both data sets. Furthermore, when we used the automatic stopping criterion associated with the Simple Margin technique, the active learner stopped requesting new labels after about 100 samples in the artifacts data set and after about 150 samples in the EDR data set (1/6th and 1/5th the number of samples required by the passive learner, respectively). Furthermore, even though the active learner required fewer labels than the passive learner, very similar held-out test accuracies were achieved.

\(^{3}\)The full paper is reproduced in Appendix B
Chapter 5

Point Process Modeling

5.1 Background

While much work has been done in the lab to figure out why and when electrodermal responses (EDR) happen, less is known about these responses in ambulatory monitoring especially during sleep. Spontaneous electrodermal responses during sleep, or sleep storms, have been observed since the 1960s (e.g., [25, 28, 29]). Although counter-intuitive to an emotional arousal interpretation, these studies have found that sleep storms occurred most frequently during slow wave sleep and least frequently during REM sleep. However, all of these studies were conducted in sleep lab environments and for a limited number of nights.

Recently, Sano et. al. has conducted a series of studies monitoring EDA while participants slept at home or in a sleep lab for an extended period of time [37]. In addition to duplicating the results of high probabilities of sleep storms during slow-wave sleep, Sano et. al. provide several characteristics of typical EDA patterns during sleep at home, including typical numbers of EDR events detected over the night and in 30s time windows. However, this study fails to capture the more complicated dynamics of these sleep storms both within a night and across several days. Furthermore, the rate of electrodermal responses is typically captured in features like the total number of responses in a window of time; however, we are suggesting that this does not fully capture the information present in the signal because it does not have a dependence on the historical rate of EDRs.

In the computational neuroscience field, recent developments in the state-space point process\(^1\) have allowed researchers to study between-trial and within-trial neural spiking dynamics [16]. The state-space point process model was developed because other models

\(^1\)A model of random binary events occurring in continuous time with parameters that evolve over time
of systems with time-varying parameters (like the Kalman Filter) are not well suited to the problem of impulse trains because of their assumptions about the data. Since its original development, the state-space point process model has been used to model several different kinds of neuronal data (e.g., [13, 39, 43]) and heart rate data (e.g., [2, 13]). Furthermore, the state-space point process was developed to provide a framework for statistical inference. We use the state-space generalized linear model (SS-GLM) introduced by Czanner et. al. to model electrodermal responses during sleep [13]. This model was chosen for its ability to capture between-night and within-night dynamics of a point process.

In Section 5.2, we provide high-level details about how the model is defined and how models are selected from a group of candidate models. Section 5.3 discusses the procedure we used for individually fitting and selecting models for SNAPSHOT participants, including a discussion on why we chose to model EDR events during sleep. We present our results in Section 5.4.

5.2 State-Space Generalized Linear Model

A detailed description of the State-Space Generalized Linear Model is given in the Czanner et. al. paper [13]. We give only the high-level details in this section and provide derivations in Appendix C.

State-space models have been used for many years in many fields including engineering, statistics, and computer science and are sometimes called hidden Markov or latent process models. They are made up of two governing equations: the observation equation and the state equation. The observation equation models how observed events are related to the state. In this case we will use a point process that depends on both within a single night (within-night) and between multiple nights (between-night) dynamics to model our observations (i.e., a series of EDRs across several successive nights). The state equation defines an unobservable process switching between states; this equation will represent the between-night dynamics of EDRs. In the case of EDR event modeling, the state will refer to the level of EDR event detection during a time window.

5.2.1 Observation Equation

We model our data (a series of EDRs across several nights) as a point process: a time-series of random binary \{0, 1\} events (1 = EDR event) that occur in continuous time. As discussed in the Czanner et. al. paper, a point process can be completely characterized
by a conditional intensity function \[^{[13]}\]. For this model, we define a conditional intensity function at time \(t\) given the history of EDR events \((H_t)\) up to time \(t\) as \(\lambda(t|H_t)\). Given an EDR event train of duration \(T\), we define \(N(t)\) to be the number of EDR events counted in an interval \((0, t]\) for \(t \in (0, T]\). Thus, the conditional intensity function is defined as:

\[
\lambda(t|H_t) = \lim_{\Delta \to 0} \frac{Pr[N(t + \Delta) - N(t) = 1|H_t]}{\Delta} \tag{5.1}
\]

We see that the probability of a single-spike in a small interval \((t, t + \Delta]\) is approximately \(\lambda(t|H_t)\Delta\).

To simplify the model, the CIF can be defined for discrete time by choosing the number of subintervals, \(L\), to split the observation window. That is, we divide the time interval \(T\) into \(L\) subintervals of width \(\Delta = TL^{-1}\). We choose \(L\) large enough so that the subinterval contains at most 1 event. These subintervals are indexed by \(l\) for \(l \in \{1, ..., L\}\). Because we are interested in modeling how this event train evolves across several nights, we index the nights by \(k\) (where \(k \in \{1, ..., K\}\) with \(K\) being the total number of nights).

Let \(n_{k,l}\) be equal to 1 when there is an EDR event in the interval \(((l - 1)\Delta, l\Delta]\) on night \(k\) and 0 otherwise. Finally, let \(H_{k,l}\) be the history of events in night \(k\), up to time \(l\Delta\). This allows us to write a conditional intensity function for the EDR event train recorded at time \(l\Delta\) of night \(k\) as the product of the time since sleep onset component, \(\lambda^S(l\Delta|\theta_k)\), and history component, \(\lambda^H(l\Delta|\gamma, H_{k,l})\), as follows:

\[
\lambda_k(l\Delta|\theta_k, \gamma, H_{k,l}) = \lambda^S(l\Delta|\theta_k)\lambda^H(l\Delta|\gamma, H_{k,l}) \tag{5.2}
\]

where \(\theta_k\) determines the effect the time since sleep onset has on the EDR event activity (i.e., they define the between-night dynamics) and \(\gamma\) determines how the probability of an EDR event at the current time depends on the recent history of EDR events.

In particular, we model the effect of the time since sleep onset (in units of EDR events per second) on EDR events by assuming the following form for the stimulus component:

\[
\log \lambda^S(l\Delta|\theta_k) = \sum_{r=1}^{R} \theta_{k,r}g_r(l\Delta) \tag{5.3}
\]

where \(g_r(l\Delta)\) is one of \(R\) functions that models the time since sleep onset effect. The choice of \(g_r(l\Delta)\) functions is flexible, and could be polynomials, splines, or unit pulse functions. We choose to use non-overlapping unit pulse functions in order to follow Czanner et. al. \[^{[13]}\]. Additionally, this will provide the ability to compare directly with traditional peristimulus time histogram (PSTH) models that are very similar to the way researchers
typically approximate EDR event rates.

The $g_r(l\Delta)$ functions are parameterized by $\theta_k = [\theta_{k,1}, \ldots, \theta_{k,R}]$ for each night $k$; thus there is a different $\theta_k$ for each night $k$. This allows us to model how the time into the night effects the EDR event rate during different nights. The relationship of the $\theta_k$ parameters is governed by a state-space equation which we describe in Section 5.2.2.

Following Czanner et. al., we model the history component of the conditional intensity function (see Eqn. 5.2) at time $l\Delta$ as a function of history parameters $\gamma$ and the EDR event history $H_{k,l}$ as a dimensionless component. The history component takes on the following form:

$$
\log \lambda^H(l\Delta|\gamma, H_{k,l}) = \sum_{j=1}^{J} \gamma_j n_{k,l-j}
$$

(5.4)

where $\gamma = [\gamma_1, \ldots, \gamma_J]$ and $n_{k,l}$ is a binary indicator of whether an event happened at time $l\Delta$ in trial $k$. We note that because Eqn. 5.4 only requires $J$ time-steps into the past, we only need to save those $J$ binary indicators in the history variable $H_{k,l} = \{n_{k,l-1}, \ldots, n_{k,l-J}\}$. Also, we note that the EDR event parameter $\gamma$ is independent of night, $k$. Thus, this component of the conditional intensity function seeks to model the underlying biophysical properties of EDR events.

Putting Eqn. 5.2, 5.3, and 5.4 together, the conditional intensity function (or observation equation) becomes:

$$
\lambda(l\Delta|\theta_k, \gamma, H_{k,l}) = \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\}
$$

(5.5)

5.2.2 State Equation

Now that we have described the observation equation (see Eqn. 5.5), we now define the state equation as a simple random walk model following Czanner et. al. [13]. In particular, we define the equation as:

$$
\theta_k = \theta_{k-1} + \epsilon_k
$$

(5.6)

where $\epsilon_k$ is an $R$ dimensional Gaussian random vector with mean 0 and unknown covariance matrix $\Sigma$. This covariance matrix will be estimated using Estimation Maximization techniques described below. We note that smaller values of $\Sigma$ will give smaller average differences between stimulus components of successive nights ($\theta_k - \theta_{k-1}$).
5.2.3 Estimation of Model Parameters using Expectation Maximization

Now that we have both an observation equation (Eqn. 5.5) and a state equation (Eqn. 5.6), we need a method to estimate the unknown parameter $\psi = (\gamma, \theta_0, \Sigma)$. As pointed out by Czanner et. al., because $\theta_k$ is unobservable and $\psi$ is unknown, we use the Expectation Maximization (EM) algorithm to compute their estimates. The derivation of this algorithm for our set up of this model is given in Appendix C.

This algorithm gives us the parameter estimates of $\theta_{k|K}$ and $\hat{\psi} = (\hat{\gamma}, \hat{\theta}_0, \hat{\Sigma})$.

5.2.4 Goodness-of-Fit

Because standard distance measures are not suitable for point-process data, we use the time re-scaling theorem to transform the point-process EDR event data into a continuous measure so that we can apply appropriate goodness-of-fit tests\[8, 32\]. The re-scaled times, $z_{k,m}$ are computed for each EDR event $m$ in each night $k$ as follows:

$$
    z_{k,m} = 1 - \exp \left[ - \int_{u_{k,m-1}}^{u_{k,m}} \lambda(u|\theta_{k|K}, \hat{\gamma}, H_{k,u})du \right]
$$

(5.7)

where $u_{k,m}$ is the time of EDR event $m$ in night $k$. If the conditional intensity function is a good approximation to the true conditional intensity of the point process, the re-scaled times $z_{k,m}$ will be independent, uniformly distributed random variables on the interval $[0, 1)$. We note that because Eqn. 5.7 is one-to-one, we can directly compare the rescaled times to a uniform distribution to assess the goodness-of-fit of the model.

First, we evaluate whether the $z_{k,m}$ are uniformly distributed by ordering them from smallest to largest and denoting the ordered values as $z_{m^*}$ where $m^* = 1, ..., M$ with $M$ being the total number of observed spikes. We then plot these ordered re-scaled times $z_{m^*}$ against the quantiles of the cumulative distribution function of the uniform distribution on $[0, 1)$. We note that the quantiles are defined as $b_{m^*} = (m^* - 0.5)/M$. Czanner et. al. term this plot as a Kolmogorov-Smirnov (KS) plot and note that if the model is consistent with the data, then the points should lie on the $45^\circ$ line. We note that the Kolmogorov-Smirnov statistic for our model is the supremum of the (vertical) distances between the K-S plot and the $45^\circ$ line. Thus, smaller K-S statistics indicate a better model fit.

Next, we evaluate whether the $z_{k,m}$ are independent by taking advantage of the structure of the model and assessing the independence of the Gaussian transformed re-scaled
times \((z^G_{k,m})\) by analyzing their autocorrelation structure. In particular, we transform \(z_{k,m}\) using the cumulative distribution function of a standard Gaussian random variable, \(\Phi(\cdot)\), as follows:

\[
z^G_{k,m} = \Phi(z_{k,m})
\]

(5.8)

We can then determine the degree to which the Gaussian transformed re-scaled times, \(z^G_{k,m}\), are independent by plotting their autocorrelation function (ACF). We can then count the number of ACF values that are outside a 95% confidence interval around zero to approximate the independence of the re-scaled times up to lag 100. We note that smaller numbers of ACF values outside this confidence interval indicates a better model fit.

5.2.5 Model Selection

Because there are several hyper-parameters that can be set (e.g., number of lagged history parameters \(J\) and the number of unit impulse functions \(R\)), we can fit several models to the EDR event data. We use Akaike’s Information Criterion (AIC) to identify the best model [9]. The AIC score is a combination of the log likelihood of the model and is penalized for higher numbers of parameters; it is computed as follow:

\[
\text{AIC}(p) = -2 \log P(N|\psi) + 2p
\]

(5.9)

where \(p\) is the number of parameters of the model and \(\log P(N|\psi)\) is the log likelihood of the observed data. Thus, in theory the model with the smallest AIC score is the best model; however, following Czanner et. al., we acknowledge that small fluctuations in AIC scores appear. Therefore, if two models have AIC scores within 10, we choose the model with fewer parameters.

5.2.6 Special Cases of the State-Space Generalized Linear Model

While we are evaluating the hypothesis that the EDR event trains during sleep rely on both between-night and within-night dynamics, it is possible that the data are better modeled in a simpler way than the proposed state-space generalized linear model (SS-GLM).

In the simplest case, we could assume that effect of the time since sleep onset is the same for all nights and that there is no dependence on EDR event history; that is, we could assume that \(\theta_k = \{\theta_{k,1}, ..., \theta_{k,R}\} = \theta = \{\theta_1, ..., \theta_R\} \) for all \(k\) and \(\lambda^H(l\Delta|\gamma, H_{k,l}) = 1\) for all \(l\Delta\). The conditional intensity function would become:
\[ \lambda_k(l\Delta|\theta) = \exp \left\{ \sum_{r=1}^{R} \theta_r g_r(l\Delta) \right\} \]  \hspace{1cm} (5.10)

Note that the conditional intensity function, \( \lambda_k(l\Delta|\theta) \), is the same for all nights \( k \).

Furthermore, since we have chosen the functions \( g_r(l\Delta) \) to be non-overlapping, equal width unit pulse functions, this model has become an inhomogeneous Poisson Process and is equivalent to a peristimulus time histogram (PSTH). The \( \theta \) parameters can be easily estimated by counting the number of EDR events that occurred during the time determined by the unit pulse functions. In particular, \( \theta_r \) is the number of EDR events that occurred between times \( l\Delta = [(r-1)L-1 + 1]\Delta \) and \( l\Delta = [rLR^{-1}]\Delta \) summed across all nights.

It is worth noting that this PSTH model is what is commonly used in EDR event modeling currently; that is, typical EDR event modeling estimates the rate of EDR events using only counts of these events without considering how the rate might be effected due to recent EDR events or what happened during a previous night.

The other special case of the SS-GLM allows there to be a history dependence to the EDR event rate, but still assumes that the effect of time since sleep onset is the same for all nights \( (\theta_k = \{\theta_{k,1}, ..., \theta_{k,R}\} = \theta = \{\theta_1, ..., \theta_R\} \) for all \( k \)). Thus, the conditional intensity function for this model has the form:

\[ \lambda_k(l\Delta|\gamma, \theta, H_{k,l}) = \exp \left\{ \sum_{r=1}^{R} \theta_r g_r(l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \]  \hspace{1cm} (5.11)

Thus, unlike the PSTH model, this model has a different conditional intensity during different nights because it depends on the EDR event history of that night. It can therefore capture between-night dynamics, but only those due to different EDR event trains. We refer to this model as the Static Generalized Linear Model (Static-GLM) with a log-link, and it can be solved using fitting procedures in various out-of-the-box solvers.

### 5.3 Modeling EDR events during sleep

After collecting 1000s of hours of electrodermal activity data from 164 participants over the course of 30 days of home-sleep during the SNAPSHOT study (see Chapter 2 for more details), we then used our binary artifact detection algorithm described in Chapter 4 to detect artifacts for all collected data. Thus, every 5-second period of electrodermal activity data was labeled as containing artifacts or containing no artifacts (see Figure 5.1 and 5.2 for
examples). We observe that many more artifacts are detected during wake periods which makes sense because individuals are moving and adjusting their wearable sensor (the most common causes of EDA artifacts) more while awake.

![Figure 5.1: Detected electrodermal activity artifacts have been labeled in red. This data set comes from a SNAPSHOT participant during sleep. The noise in this example is most likely due to the participant moving during sleep and the pressure on the Q-sensor changing resulting in a different skin conductance measurement.](image1)

Figure 5.1: Detected electrodermal activity artifacts have been labeled in red. This data set comes from a SNAPSHOT participant during sleep. The noise in this example is most likely due to the participant moving during sleep and the pressure on the Q-sensor changing resulting in a different skin conductance measurement.

![Figure 5.2: Detected electrodermal activity artifacts have been labeled in red. This data set comes from a SNAPSHOT participant during wake. Here the noise is likely due to the participant walking around or moving his or her wrist while talking, writing, or working on a computer.](image2)

Figure 5.2: Detected electrodermal activity artifacts have been labeled in red. This data set comes from a SNAPSHOT participant during wake. Here the noise is likely due to the participant walking around or moving his or her wrist while talking, writing, or working on a computer.

In addition to observing more artifact-free data during sleep, we hypothesized that EDR events would be more consistent in timing from night-to-night given the previous literature on EDA during sleep. For these reasons, we decided to continue our analysis looking at EDR events during sleep only. We used a combination of self-reported sleep time cross-validated by an expert with access to the participant’s actigraphy data from the Actiwatch in order to determine when the participant was asleep (see Chapter 2 for more information).

Using the EDR event detection method described in Section 3.2, we detected EDR events for each night (see Figure 5.3 for an example of one participant’s detected EDR
events in an Sleep EDR Raster Plot). This signal was then transformed into a binary (EDR event detected= 1) 1Hz signal for all further processing. This sampling rate was chosen to be as large as possible without containing 2 EDR events in one time window so that we could apply the signal to the state-space generalized linear model described in Section 5.2 (see Figure 5.4 for a distribution of inter-event interval (IEI) times).

Figure 5.3: EDR Sleep Raster Plot which plots one vertical line for each detected EDR event for one participant during sleep

After using our artifact detection for every sleeping period, we computed the percentage of artifact-free data for each sleep period. Then, for each participant, we computed the median of their artifact-free percentages in order to get a rough estimate of which participants had higher quality data. We kept all participants who had a median score of greater than 0.6, or 60% (see Figure 5.5 for a distribution of median percentages). We also discarded participants who had 3 or more days where no EDR events were detected during sleep (see Figure 5.6 for a distribution of number of nights without any EDRs detected).

\footnote{We note that 1Hz (or $\Delta = 1s$) was chosen so that it was as long as possible while still being faster than any two consecutive EDRs we observed.}
Figure 5.4: Left-most tail of distribution of inter-event interval (IEI) times for one SNAP-SHOT participant, computed as the number of seconds between successive EDR events. The minimum IEI was 1 second, the maximum was 7738 seconds, the median was 11.0 seconds and the mode was 6 seconds for this participant. Other participants had similar distributions.

Figure 5.5: Distribution of Participants’ Median of Percentage of Artifact-Free Data with the dropping threshold marked in red.

This resulted in dropping 102 participants so that we were left with 62 participants. This was done to remove participants that were likely wearing their devices too loose or
had several days of missing data. We then dropped an additional 16 participants for the
PSTH and Static-GLM models and an additional 8 participants for the SSGLM model so
that each modeled participant had enough EDR events so that we could compute goodness-
of-fit tests properly. This left 46 participants for the PSTH and Static GLM models and 38
participants for the SSGLM models. However, the discarded participants did not appear to
be biased based on gender, sleep regularity, sleep quality (as determined by PSQI score),
mental health (as determined by MCS score), alcohol consumption, or phone usage.

For each participant, we split his or her data into 3 parts using every 3rd day so that we
could preserve the time component that we are seeking to model. We then took the first
third of the data set for each participant (i.e., the 1st, 4th, 7th, etc. days) and estimated
parameters for a range of parameters for all three types of models (PSTH, Static-GLM, and
SS-GLM). Although splitting the data into thirds will reduce the explanatory power of the
model, it will allow us to check how well the model explains held-out test data. Once we
select the best model, we will re-train the model using all the data before interpreting the
parameter estimation and inference results. Results of the model estimation and inference
are given in Section 5.4

For the PSTH models, the only hyper parameter we varied was the size of the unit
impulse functions, $g_r(l \Delta)$. However, for the Static-GLM and SS-GLM, we also varied the
lag components. Recall that the difference between Static-GLM and SS-GLM models is

![Figure 5.6: Distribution of Number of Sleep Episodes with No Detected Peaks with the
dropping threshold marked in red.](image-url)
that the Static-GLM model assumes that the time since sleep onset effect is the same across all nights, but the SS-GLM model doesn’t make this assumption. Thus, we do not need to vary any additional hyper parameters for the SS-GLM model because the difference is in the estimation of the parameters.

To make the problem slightly more tractable, we constrained the lag components to be binned so that the $\gamma_j$ parameters are the same for certain time bins. For example, we could have each $\gamma_j$ be computed independently, or we could constrain the $\gamma_j$’s so that every 3 parameters had to be the same (i.e., $\gamma_1 = \gamma_2 = \gamma_3$ and $\gamma_4 = \gamma_5 = \gamma_6$, and so forth); we opted for the latter.

We fit models for 0.5, 1.0, 1.5, 2.0, and 3.0 hour $g_r(l\Delta)$ lengths and for two types of lags:

1. **Equal Bins:** 5 second bins for 20 to 180 seconds prior to the current time (e.g., 1–5, 6–10, 11–15, 16–20, 21–25, 26–30)

2. **Irregular bins:** 2 second bins for 1 to 20 seconds and then 10 second bins until 180 seconds prior to the current time (e.g., 1–2, 3–4, 5–6, 7–8, 9–10, 11–12, 13–14, 15–16, 17–18, 19–20, 21–30, 31–40)

Because of the complexity of the SS-GLM model and the number of participants we modeled, we only fit models for 2.0 and 3.0 hour $g_r(l\Delta)$ lengths and only irregular bins after finding that these models outperformed models with shorter lengths and models with equal bins.

For each model, we computed the goodness-of-fit metrics and AIC score and then we chose the best lag parameters for each length of $g_r(l\Delta)$ functions based on minimal AIC scores for each model type. Then, for each of the best models, we evaluated the goodness-of-fit metrics on the held-out 2nd and 3rd datasets for each participant in order to determine how well the model would fit unseen data.

We then selected 11 participants with SS-GLM models that did better than Static-GLM models in terms of AIC scores and K-S statistic to do further analysis on. These selected participants were all male except 1 participant. Additionally, these 11 participants had higher sleep regularity, self-reported that they were calmer in the evenings, and had higher mental health (as determined by MCS scores). We hypothesize that it was easier to fit a model to regular sleepers because their EDR patterns were more regular. Also, from previous work, we know that participants in our data set that had higher sleep regularity were more likely to be healthier.

---

3We note, however, that only the average evening calmness was significant with $p < 0.05$. 

However, it is important to note that these 11 selected participants had similar sleep quality (as determined by PSQI scores) and alcohol consumption distributions as the dropped participants. Additionally, these selected participants had similar clean percentages and a similar number of detected EDRs as the dropped participants.

We are particularly interested in doing statistical inference on these 11 participants. Therefore, we retrained the previously selected best SS-GLM model using all days of the data and then used maximum-likelihood and Monte-Carlo methods described in Czanner et. al. to make inferences about the structure of the EDR event data during sleep [13].

5.4 Results

After each participant’s first third of data was modeled as described in Section 5.3, we computed the AIC and goodness-of-fit metrics in order to compare models. What follows is analysis of which models were chosen for different participants to see if there were systematic differences between different types of people. Then we will discuss how the best models fit the held-out data. Finally, we will review the inferences that can be made with the SS-GLM model when trained with the full data set.

5.4.1 Selected Models

In Tables 5.1, 5.2, and 5.3 we can see the number of participants for which the best model (as defined as Minimum AIC score) was a certain parameter selection.

<table>
<thead>
<tr>
<th>Time Window Length</th>
<th>Number of Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5 hours</td>
<td>12</td>
</tr>
<tr>
<td>1.0 hours</td>
<td>2</td>
</tr>
<tr>
<td>1.5 hours</td>
<td>8</td>
</tr>
<tr>
<td>2.0 hours</td>
<td>13</td>
</tr>
<tr>
<td>3.0 hours</td>
<td>11</td>
</tr>
</tbody>
</table>

For the PSTH models, we observe that the participants are spread over the time window length parameter. However, in the Static GLM models the longer Time Window Lengths were chosen. We note that minimum AIC score was used to select Time Window Length and Maximum Lag Time. This is why each participant has a best “Equal Bin” and “Irregular Bin” for the Static GLM models. It is worth noting, however, that only 2 participants
Table 5.2: Selected Static GLM Models using Minimum AIC Score

<table>
<thead>
<tr>
<th>Lag Type</th>
<th>Time Window Length</th>
<th>Max Lag</th>
<th>Number of Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equal Bins</td>
<td>1.5 hours</td>
<td>20 s</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>30 s</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>2.0 hours</td>
<td>20 s</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>40 s</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3.0 hours</td>
<td>20 s</td>
<td>13</td>
</tr>
<tr>
<td></td>
<td></td>
<td>30 s</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td></td>
<td>40 s</td>
<td>2</td>
</tr>
<tr>
<td>Irregular Bins</td>
<td>2.0 hours</td>
<td>20 s</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td>30 s</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>3.0 hours</td>
<td>20 s</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>30 s</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td></td>
<td>40 s</td>
<td>13</td>
</tr>
</tbody>
</table>

Table 5.3: Selected SS-GLM Models using Minimum AIC Score

<table>
<thead>
<tr>
<th>Time Window Length</th>
<th>Max Lag</th>
<th>Number of Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0 hours</td>
<td>20 s</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>20 s</td>
<td>12</td>
</tr>
<tr>
<td></td>
<td>30 s</td>
<td>6</td>
</tr>
<tr>
<td>3.0 hours</td>
<td>40 s</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>50 s</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>60 s</td>
<td>13</td>
</tr>
</tbody>
</table>

had better models with the “Equal Bins” (using minimum AIC) than “Irregular Bins.” It is also interesting to note the selected Max Lag Time in the Static GLM models was between 20 and 40 seconds for all participants even though we fit models for up to 180 second lags. This could indicate that there is a general similarity in physiological processes that govern electrodermal activity between different people. Finally, we note that all but 2 participants selected SS-GLM models with a Time Window Length of 3 hours. Here we do see a wider spread of Max Lag Times selected with extra weight on the tails of 20s and 60s.

If we compare the 3 models against each other using the median values of AIC, K-S statistic, and number of ACF points significantly different than zero, we see that the general trend is that the more sophisticated the model, the better the model fit (see Table 5.4). However, we point out that there was a wide spread in the SS-GLM models; that is, there were several participants for whom the SS-GLM was a very bad model fit. We compare the model ranking for each participant (see Table 5.5) and note that no participant has the

---

4 Recall that lower numbers for each of these metrics denotes a better model fit.
PSTH model as the best model, but the PSTH model is sometimes better than the SS-GLM model (using AIC and K-S Statistic). Thus, the traditional EDR modeling method of simply counting the number of EDR events in a time window does not capture the patterns in the EDR event data like the SS-GLM and Static-GLM models. Specifically, it lacks the ability to capture the effect of recent history.

Table 5.4: Comparing Models using AIC and Goodness-of-Fit Metrics

<table>
<thead>
<tr>
<th>Model Type</th>
<th>AIC</th>
<th>K-S statistic</th>
<th>ACF points</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSTH</td>
<td>9140</td>
<td>0.548</td>
<td>23</td>
</tr>
<tr>
<td>Static-GLM</td>
<td>7769</td>
<td>0.283</td>
<td>8.5</td>
</tr>
<tr>
<td>SS-GLM</td>
<td>7183</td>
<td>0.231</td>
<td>29.5</td>
</tr>
</tbody>
</table>

Table 5.5: Number of Participants with each Model Ranking Order based on AIC and K-S Statistic. We also show the number of participants that satisfy ordering based on both metrics at the same time

<table>
<thead>
<tr>
<th>Model Ranking (Best to Worst)</th>
<th>AIC</th>
<th>K-S statistic</th>
<th>Both</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-GLM Static-GLM PSTH</td>
<td>21</td>
<td>22</td>
<td>11</td>
</tr>
<tr>
<td>SS-GLM PSTH Static-GLM</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Static-GLM SS-GLM PSTH</td>
<td>9</td>
<td>12</td>
<td>3</td>
</tr>
<tr>
<td>Static-GLM PSTH SS-GLM</td>
<td>8</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>PSTH SS-GLM Static-GLM</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PSTH Static-GLM SS-GLM</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

5.4.2 Model Fit on Held-out Data

After selecting the best models based on minimum AIC scores, we computed the K-S statistic (a Goodness-of-Fit metric) on the held-out data. In Figure 5.7, we notice that the Static-GLM models do significantly worse on the held-out data, but perform about the same on either held-out set of the data. In Figure 5.8, we see that the K-S statistics on the held-out data in the SS-GLM models is much more similar to the K-S statistics of the training data when compared to the Static-GLM models. This is most likely due to the SS-GLM model capturing the between-night dynamics directly with the state equation.

When we ran an ordinary least squares regression for predicting sleep regularity using the K-S Statistic of the training data for the SS-GLM model, we found that the coefficient

5However, the held-out K-S statistics are still significantly ($p < 0.05$) higher than the training data
6We removed one individual because he was an outlier with a very high sleep regularity and high K-S Statistic. This individual is labeled in red on the plot.
Figure 5.7: After training the best Static-GLM model for each participant based on the 1st third of the data, we computed the K-S Statistic for the 2nd and 3rd thirds of the data. We note that (b) is broken out by Time Window Length and that the trend of significantly worse K-S statistics on the held-out data is still present.

on the K-S Statistic $-15.4$ was significantly different with $p < 0.05$ (see Figure 5.9). Thus, the worse the model fit (i.e., a higher K-S Statistic), the lower the sleep regularity.

Because the SS-GLM is fitting held-out data better than the Static-GLM modes and because we are interested in doing statistical inference with the SS-GLM model, we select
Figure 5.8: After training the best SS-GLM model for each participant based on the 1st third of the data, we computed the K-S Statistic for the 2nd and 3rd thirds of the data. We note that (b) is broken out by Max Lag for comparison (refer to Table 5.3 for number of participants in each bin).

11 participants to be analyzed for inference (see Table 5.6 for a list of participants and the selected models, AIC scores, and Goodness-of-Fit statistics). These participants were selected because they had lower AIC and K-S Statistics for the SS-GLM models when compared with their Static-GLM models.
5.4.3 History Effects

Using the models from the final 11 participants (trained on all 30 days of data), we can look at patterns in the history effect (see Figure 5.10). We observe that there is a suppression of EDR activity 1-2 seconds after an EDR event, but that there is an increase in likelihood that an EDR event would happen 3-4 seconds after an event. This increase in likelihood is continued, but at a decreasing rate for the rest of the time bins. This decrease in likelihood right after an EDR event seems to follow the rule of thumb that EDR events are separated by at least 1 second [7].

5.4.4 Time Since Sleep Onset Results

Another parameter that is estimated in the SS-GLM model is the time since sleep onset effect \( \exp \{ \theta_{k|K} \} \) which gives a multiplicative factor for each time window (reds indicate an increase in probability of an EDR event detection and blues indicate a decrease). In Figure 5.11 we see two participant’s estimated \( \theta_{k|K} \) values for each time window and for each day in the study. We can see that the participant in Figure 5.11a had milder changes between successive days in the study; however, we see that the participant in Figure 5.11b usually has a decreased probability except days 0,2, and 11 where there is a marked increase. The other 9 participants also have unique time since sleep onset effects;
Table 5.6: Final Selected SS-GLM Models to be Analyzed for Inference

<table>
<thead>
<tr>
<th>Time Window</th>
<th>Max Lag</th>
<th>PPT</th>
<th>AIC</th>
<th>K-S statistic</th>
<th>ACF Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.0 hours</td>
<td>20 s</td>
<td>CS13M014</td>
<td>7507.8197</td>
<td>0.222637</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS14M027</td>
<td>3795.2657</td>
<td>0.112890</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS14M074</td>
<td>2780.2032</td>
<td>0.222573</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS14M098</td>
<td>5717.1249</td>
<td>0.151164</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS15M033</td>
<td>2340.3209</td>
<td>0.278565</td>
<td>4</td>
</tr>
<tr>
<td>30 s</td>
<td></td>
<td>CS14M055</td>
<td>2663.5215</td>
<td>0.207203</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS14M078</td>
<td>7862.8520</td>
<td>0.170717</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS15M010</td>
<td>8347.4044</td>
<td>0.116680</td>
<td>7</td>
</tr>
<tr>
<td>40 s</td>
<td></td>
<td>CS13M003</td>
<td>6067.3106</td>
<td>0.261566</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS15M024</td>
<td>4324.4092</td>
<td>0.133495</td>
<td>5</td>
</tr>
<tr>
<td>60 s</td>
<td></td>
<td>CS15M008</td>
<td>6221.2695</td>
<td>0.152799</td>
<td>8</td>
</tr>
</tbody>
</table>

Figure 5.10: For each time bin on the x-axis, we plot a boxplot of the history effect ($\gamma$ values) across 11 participants. Parameters above (below) the red line increase (decrease) the conditional intensity function when an EDR event is detected in history during that time bin. Refer to Table 5.6 for the number of participants that had history parameters for each time bin.

therefore, unlike the history effect, there appears to be no global pattern of time since sleep onset effects.

Using Monte-Carlo techniques described in Czanner et. al. [13], we can compute a comparison between EDR event rates between pairs of time periods during sleep. In particular, we can compute the probability that the rate during one period is greater than another.
period within a night. If the probability of the rate of period 1 is greater than the rate of period 2 is above 95% for night \( k \), then we can say that the rate of period 1 has a high probability of being greater than the rate of period 2. Note that this is not a significance test because we are computing empirical Bayes’ estimates of the probabilities.

We first compared the first 90 minutes to the second 90 minutes after sleep onset. For each participant, we computed the rate during the first 90 minutes and during the second 90 minutes for each night. Then, for each participant, we computed the percentage of nights that the rate during the first 90 minutes had a 95% probability of being greater than the second 90 minutes and vice-versa. Thus, for each person we have a percentage for higher rates during the first 90 minutes and a percentage for higher rates during the second 90 minutes. We note that these two percentages don’t add to 100% because there may be nights when the probability of the rates being different are less than 95%. We also computed for each participant the percentage of nights that the rate during the first 3 hours was greater than the second 3 hours and vice-versa.

When comparing the first two 90 minutes of sleep, we found for 8 of the 11 participants there were a high percentage of nights (\( > 80\% \)) that one of the two 90 minute periods had a high probability of being greater than the other (see Table 5.7). However, we found that only 2 participants had this same pattern when comparing the first 3 hours and the second 3 hours. In other words, when comparing the first 90 minutes and second 90 minutes, one of the two will likely have a high probability of one rate being greater than the other; when comparing the first 3 hours and the second 3 hours, they are more likely to have similar rates.

Furthermore, we found that 8 of 11 participants had more days with higher rates during the second 90 minutes than the first. This seems to agree with Sano et. al. finding that the onset of the first “sleep storm” (or period of high EDR events) was most probable 60 to 120 minutes after sleep onset [37]. Additionally, we found that there was 6 of 11 participants that had higher rates during the first 3 hours than during the second three hours, and therefore 5 of the 11 participants with higher rates during the second three hours. Thus, our finding seems to again agree with Sano et. al.’s 2011 finding of higher rates later in the night [37], but contradict with i Baque et. al. who found that spontaneous EDA activity is less frequent during the latter half of sleep [22]. However, Sano et. al. only compares 3 participants and notes that the more regular sleeper had higher rates earlier in the night. When we compare the regularity of participants with more days with higher earlier (0-3 hours after sleep onset) rates to participants with more days with higher later (3-6 hours after sleep onset) rates, we find that the earlier group had a higher median regularity
Table 5.7: Empirical Bayes Estimates of EDR Event Rates during Different Periods of the Night

<table>
<thead>
<tr>
<th></th>
<th>Percentage of Nights</th>
<th>0-90 min &gt;  90-180 min</th>
<th>0-180 min &gt; 180-360 min</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&gt; 95% &lt; 5%</td>
<td>&gt; 95% &lt; 5%</td>
<td></td>
</tr>
<tr>
<td>CS13M003</td>
<td>67.9% 21.4%</td>
<td>39.3% 35.7%</td>
<td></td>
</tr>
<tr>
<td>CS13M014</td>
<td>35.5% 64.5%</td>
<td>29.0% 19.4%</td>
<td></td>
</tr>
<tr>
<td>CS14M027</td>
<td>6.7% 56.7%</td>
<td>0.0% 80.0%</td>
<td></td>
</tr>
<tr>
<td>CS14M055</td>
<td>37.5% 46.9%</td>
<td>37.5% 15.6%</td>
<td></td>
</tr>
<tr>
<td>CS14M074</td>
<td>40.0% 36.7%</td>
<td>80.0% 6.7%</td>
<td></td>
</tr>
<tr>
<td>CS14M078</td>
<td>44.8% 55.2%</td>
<td>27.6% 31.0%</td>
<td></td>
</tr>
<tr>
<td>CS14M098</td>
<td>17.2% 79.3%</td>
<td>31.0% 44.8%</td>
<td></td>
</tr>
<tr>
<td>CS15M008</td>
<td>32.1% 67.9%</td>
<td>10.7% 32.1%</td>
<td></td>
</tr>
<tr>
<td>CS15M010</td>
<td>48.4% 48.4%</td>
<td>58.1% 22.6%</td>
<td></td>
</tr>
<tr>
<td>CS15M024</td>
<td>40.6% 56.3%</td>
<td>34.4% 25.0%</td>
<td></td>
</tr>
<tr>
<td>CS15M033</td>
<td>21.4% 53.6%</td>
<td>32.1% 42.9%</td>
<td></td>
</tr>
</tbody>
</table>

(73.3) than the later group (70.6) as Sano et. al. found. However, the difference is not significant. Furthermore, Sano et. al. expanded on the 2011 finding to 15 participants in a 2014 publication that found that “sleep storms” were more likely to be found during the first half of the night in healthy individuals [38].

The SS-GLM model also allows us to compare the rates on each day of the study. In particular, we are interested in how physical health relates to EDR rates during sleep. We hypothesized that EDR rates would be shifted later in the night when an individual is getting sick based on anecdotal evidence. Therefore, we compared the probability that the EDR rate during the first 3 hours of sleep was greater than the second 3 hours of sleep vs self-reported health and expected to see these measures correlate with one another. We can see in Figure 5.12 an example that this hypothesis seems to be true. That is, the probability of the first three hours being greater than the second 3 hours seems to be higher on days of higher health. However, because this is only a single participant and this is self-reported health, this observation should only be taken as a hypothesis with further analysis and studies needed to confirm the results.

5.5 Summary

In summary, we were able to use the SS-GLM model described by Czanner et. al. [13] to model EDR events during sleep allowing us to capture within-night and between-night
dynamics that traditional event counting methods cannot. After discarding participants with excessively noisy signals and participants without data on several nights, we were able to model 38 participants using the PSTH, Static GLM, and SS-GLM models. In comparing the AIC and goodness-of-fit metrics between these three model types, we found that the PSTH model (similar to the traditional event count methods) was the worst of the three models.

For most participants, we found that longer “time since sleep onset” windows resulted in a better model fit and that the best models needed only up to 60 seconds of the recent history of EDR events. Furthermore, we found that it was easier to fit SS-GLM models to participants who had higher sleep regularity. We found 11 participants who had the best AIC and goodness-of-fit metrics on the SS-GLM model and used these individuals to make inferences about EDR event rates during sleep. We found a suppression of EDR events 1-2 seconds after an EDR event and an increase in likelihood 3-4 seconds after an EDR event. This trend held for all 11 participants, but there were some individual differences in the magnitude of the effects.

Finally, we compared the rate of EDR events using empirical Bayesian estimates. We found that most participants had more days with higher rates of EDR events during the first 90 minutes of sleep compared to the second 90 minutes of sleep on the same night. Additionally, we also compared the first 3 hours to the second 3 hours of sleep and found that the differences in rate varied from individual to individual.

In the future, we plan to open-source our code used to model EDR events as a point-process so that the full pipeline from EDR and artifact detection to rate estimation can be easily accessible to other researchers.
(a) Participant CS13M014

(b) Participant CS15M033

Figure 5.11: The time since sleep onset parameter ($\theta_{k|K}$) is shown for two participants for all recorded days of the study. Each patch of color represents a $\theta_{k|K,r}$ with red patches representing time windows with higher probabilities of EDR events and blue representing a decreased probability of EDR events.
Figure 5.12: Probability that the rate of EDR events during the first 3 hours after sleep onset is greater than the rate of EDR events during the second 3 hours after sleep onset compared to self-reported morning health. While this is only a single example and more studies need to be conducted, we see that this participant has higher EDR rates earlier in the night when healthier.
Chapter 6

Conclusions and Future Work

6.1 Summary of Work

In this thesis, I have outlined the SNAPSHOT Study and have described the data from the 164 participants of the first 4 cohorts.

I then discussed the importance of electrodermal activity (EDA) in studies related to affective phenomena. In particular, I discussed how current methods of identification of electrodermal responses (EDRs) assume that the shape of the EDR is consistent for a single participant and how this assumption doesn’t seem to hold in the SNAPSHOT Study’s ambulatory setting. Therefore, we developed a simple technique to identify EDRs and extract shape features.

We also created a machine learning algorithm to identify artifacts in EDA data, which are very common in ambulatory studies due to the motion of the skin and muscles beneath the electrodes of the sensor. We then expanded this algorithm using active learning so that the human labeling effort of future labeling would be kept to a minimum. We have open-sourced our code for EDR detection, EDR feature extraction, and artifact detection so that other researchers are able to use our methods. These can be found at http://eda-explorer.media.mit.edu/.

Typical estimations of the rate of EDR events are computed by counting the number of events in a certain time window. While this technique has shown promise in its ability to describe certain affective states, it lacks the ability to determine how the rate is affected by the recent history and how the rate is a result of the time of day. I sought to close this gap and to model the rate of EDR events during sleep using the State-Space Generalized Linear Model (SS-GLM) and its special cases developed by Czanner et. al [13].
I found that regular sleepers fit the SS-GLM model better than irregular sleepers. Furthermore, I found that the SS-GLM models selected in a parameter sweep were the models with 2 or 3 hour time windows and 60 seconds or less of recent history required to model the current probability of an EDR event. The particular time window and recent history lengths varied between participants, with no pattern showing up currently. However, the pattern of history effects was fairly consistent across the SS-GLM models with the first time bin (1-2 seconds previous to the current time) being suppressed and the 2nd and 3rd time bin (3-4 and 5-6 seconds previous, respectively) having an increased probability. The time since sleep onset effects varied between participants. We began to investigate the connection between the self-reported health or emotional state of the individual and the nightly difference between the rate of EDR events in the first 3 hours of sleep and the second 3 hours of sleep; however, at this time we need to conduct significant future work in order to make strong inferences.

6.2 Implications

By creating and disseminating our automatic detection of EDRs and artifacts, we have opened the door to larger scale ambulatory studies of electrodermal activity and affective phenomena, which has previously required a significant amount of manual effort on the part of the researchers involved.

Furthermore, we have introduced modeling electrodermal responses as a point process by using models developed for estimating the rate of neuron spike trains. With this new modeling technique, it is possible that we can improve on our estimations of wellbeing [23, 24], which have used old techniques for estimating EDR event rates.

This model can have many applications, including providing a framework for understanding how different behavioral or environmental factors affect the physiological patterns of sleep without needing to study participants in an invasive sleep lab environment. In the long term, a successful implementation of the SS-GLM could be used to help influence behaviors towards better sleep and performance. In particular, this model could be used to help those who are already ill to adjust their behaviors in small ways in order to improve their sleep. More generally, a successful model of EDR events as a point process could provide a framework for future at-home studies of sleep and behavior.
6.3 Future Work

In the future, we would like to adapt the SS-GLM model to more accurately fit EDR event data. Unlike neuron spike trains, EDR event data tends to have very long periods of time where no events are detected. Additionally, the SS-GLM model was initially developed for specific lengths of time defined by a study protocol, but the sleeping schedule of our undergraduate population can vary widely. Therefore, it would be useful to adapt the model, perhaps by changing the “time since sleep onset” functions \( g_r(\Delta) \) to have lengths proportional to the total sleep time instead of requiring them to be the same size. Furthermore, we plan to open source the code used to model EDR event data as a state-space based point process so that the full pipeline of EDR detection, artifact removal, and EDR modeling is available.

Finally, studying the rate of EDR events in different populations would be interesting, as our population is composed of young and relatively healthy undergraduates. Thus, studies involving participants with a clinical diagnosis of mental or physical illness in comparison with healthy individuals ought to result in stronger signals. Furthermore, simultaneously conducting a polysomnography (PSG) during each night would likely yield data that would be helpful in adding to the literature on the rate of EDR events and different sleep stages.
Appendix A

Automatic Identification of Artifacts in Electrodermal Activity Data

This paper was published in the proceedings of the 37th Annual International Conference of the IEEE Engineering and Medicine and Biology Society which was held in August, 2015 [42].
Automatic Identification of Artifacts in Electrodermal Activity Data

Sara Taylor1∗, Natasha Jaques 1∗, Weixuan Chen1, Szymon Fedor1, Akane Sano1 and Rosalind Picard1

Abstract—Recently, wearable devices have allowed for long term, ambulatory measurement of electrodermal activity (EDA). Despite the fact that ambulatory recording can be noisy, and recording artifacts can easily be mistaken for a physiological response during analysis, to date there is no automatic method for detecting artifacts. This paper describes the development of a machine learning algorithm for automatically detecting EDA artifacts, and provides an empirical evaluation of classification performance. We have encoded our results into a freely available web-based tool for artifact and peak detection.

I. INTRODUCTION

Electrodermal Activity (EDA) refers to the electrical potential on the surface of the skin [1]. When the body responds to stress, temperature, or exertion, the sympathetic nervous system (SNS) increases sudomotor innervation, causing EDA to increase and perspiration to occur. Because the SNS is influenced by the hypothalamus and limbic system — structures in the brain that deal with emotion — EDA has frequently been used in studies related to affective phenomena and stress (e.g. [5], [6], [7], [8], [10], [12], [14]).

Despite its popularity, little research has been done into detecting noise and artifacts in an EDA signal. This is especially problematic given the increasing number of studies that are collecting ambulatory EDA data over long time periods using wearable devices (e.g. [2] [5] [7] [11] [14]). While these studies may provide profound insight into how affect and stress interact with other factors in daily life, continuous and unobtrusive measurement of EDA using wearable devices makes the signal collected vulnerable to several types of noise. Artifacts can be generated from electronic noise or variation in the contact between the skin and the recording electrode caused by pressure, excessive movement, or adjustment of the device. If these artifacts remain in the signal when it is analyzed they can easily be misinterpreted and skew the analysis; for example, they may be mistaken for a skin conductance response (SCR) (a physiological reaction that may indicate increased stress).

Consequently, many researchers are forced to manually inspect the data in order to decide which portions are too noisy to retain (e.g. [3]). This approach cannot scale to the type of large-scale EDA studies that are currently being proposed [7], which may involve data collected from hundreds of participants over weeks or months. In order to make collecting EDA viable in these types of studies, an automated method for detecting and removing noise and artifacts must be developed. In this paper we describe the development of both a classification algorithm for automatically detecting artifacts, and an online system hosted at eda-explorer@media.mit.edu that will apply the algorithm to users’ uploaded EDA files in order to provide them with an analysis of which portions contain artifacts.

II. RELATED WORK

Through extensive research into the physiological processes underlying EDA, as well as the electrical properties of the recording equipment used in measurement, Boucsein [1] is able to provide a complete description of the characteristic shape of an SCR: the response typically lasts between 1-5 seconds, has a steep onset and an exponential decay, and reaches an amplitude of at least .01µS (see Fig. 1 for an example of a typical SCR). However, despite the availability of this knowledge, no accepted technique for removing signal artifacts has been developed.

Fig. 1. Shape of a typical SCR

Currently, many researchers deal with signal artifacts and noise by simply applying exponential smoothing (e.g. [6]) or a low-pass filter (e.g. [8] [9] [12]). While these techniques are able to smooth small variations in the signal, they are not able to compensate for large-magnitude artifacts that can result from pressure or movement of the device during ambulatory recording. Fig. 2 shows a portion of signal that contains three obvious artifacts, in which the sharp decreases could not possibly be produced by human physiology. As is evident from comparing the raw and filtered versions of the signal, the low-pass filter has not removed the artifacts, and any subsequent analysis based on the filtered signal is likely to mistake the artifacts as genuine physiological responses.

Other researchers have used Boucsein’s analysis to develop heuristic techniques for removing atypical portions of the EDA signal. Kocielnik and colleagues [8] chose to discard portions of their data where the signal increased more than 20% per second or decreased more than 10% per
second. They verified that this approach removed artifacts based on visual inspection. Using a similar approach, Storm and colleagues manually set thresholds for the maximum and minimum amplitude, maximum slope, and minimum width of an SCR, and discarded responses that did not fit these criteria [13]. In another case, a study which collected EDA from two sensors (on both the ankle and the wrist) was able to detect artifacts by looking for epochs when only one of the two sensors had an abnormally low signal, or showed an unusually rapid increase or decrease [5].

These heuristic thresholds were developed for particular studies and participants, and verified only through visual inspection by the researchers conducting them; they may not generalize beyond those contexts. We seek to develop an empirically validated automatic technique for removing artifacts in EDA signals.

III. METHODS

In order to validate our automatic artifact detection method, we needed to establish a ground truth for what portions of an EDA signal are considered clean, and what portions contain artifacts. To do this we had two expert EDA researchers label 5-second epochs of EDA data collected from a previous experiment [3]. The labeled data was used as input to our machine learning classifier.

A. Data Collection

The data used in this analysis were collected during a study in which 32 participants completed physical, cognitive and emotional tasks while wearing Affectiva Q EDA sensors on both wrists [3]. The Q sensor collects EDA data by measuring skin conductance (SC) in microSiemens (µS) at a frequency of 8Hz. All experimental procedures were approved by the Institutional Review Board for human subjects research at MIT.

B. Expert Labeling

We created a data set of 1560 non-overlapping 5-second epochs of EDA data, sampled from portions of data that were identified as possibly containing artifacts, true SCRs, or static skin conductance level (SCL). As part of our website, we built an interface to allow our two experts to review these epochs and assign a label of either ‘artifact’ or ‘clean’. Both experts agreed on a set of criteria that defines an artifact in the signal, which is as follows:

- A peak which does not show exponential decay, depending on the context (e.g. if two SCRs occur close together in time, the first response may not decay before the second begins, yet this is not considered an artifact)
- Quantization error with ≥ 5% of signal amplitude
- A sudden change in EDA correlated with motion
- A SCL ≤ 0

Although our classification labels were created using these criteria, our website provides the ability for other researchers to agree to label their own data according to their individual application needs. The site allowed the experts to view both the raw signal and a filtered signal (to which a standard 1Hz low-pass filter had been applied), as well as the accelerometer data, which is simultaneously collected by the Q sensor. We felt that viewing the accelerometer data might help the experts to identify motion artifacts. However, we do not provide accelerometer data to our classification algorithm, for two reasons. Firstly, by training the classifier using only EDA data, we enable it to be applied to EDA signal collected from devices other than the Q that do not collect accelerometer data. Secondly, while it would be simple to discard portions of the signal with high power in the corresponding accelerometer data, this is not always desirable; for example, in applications such as detecting epileptic seizures, strong accelerometer signal occurs simultaneously with high EDA, but the EDA signal is both clean and valuable to the analysis [9]. Because we allowed the raters to skip epochs if they did not wish to label them, we eventually obtained 1301 data points that were labeled by both experts. The percentage agreement was 80.71%, and the Cohen’s $\kappa = 0.55$.

There are multiple ways to deal with epochs for which the raters’ labels did not agree. The first is to discard them, which is reasonable in the sense that we cannot establish a ground truth value for those epochs, meaning we have no way to train or assess the performance of the classifier. The second technique is to treat disagreements as a third class in which we are unsure whether the signal is clean or an artifact. We will present results from both approaches. Table I gives the datasets for both.

<table>
<thead>
<tr>
<th>Classifier</th>
<th># Clean Epochs</th>
<th># Questionable Epochs</th>
<th># Artifact Epochs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary</td>
<td>798</td>
<td>NA</td>
<td>252</td>
</tr>
<tr>
<td>Multiclass</td>
<td>798</td>
<td>251</td>
<td>252</td>
</tr>
</tbody>
</table>
C. Feature Extraction

We extracted several features for each five second epoch. Given the importance of the shape of an SCR, we began by including statistics related to the amplitude and first and second derivative of the EDA signal (see Table II). These features were computed for both the raw and filtered signal; we are not concerned about including too many features at this stage, because we later apply a feature selection procedure to reduce the chance of overfitting.

<table>
<thead>
<tr>
<th>Category</th>
<th>Specific Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw SC</td>
<td>amplitude: mean</td>
</tr>
<tr>
<td></td>
<td>1st derivative, 2nd derivative: max, min, max of absolute value, mean absolute value</td>
</tr>
<tr>
<td>Filtered SC</td>
<td>max, mean, standard deviation, median, number aboveZero</td>
</tr>
<tr>
<td>Wavelet coefficients</td>
<td></td>
</tr>
</tbody>
</table>

We then used a Discrete Haar Wavelet Transform to compute additional features that may be indicative of sudden changes in the EDA signal. Wavelet Transforms have been successfully used in several noise reduction applications; because of their good time-frequency localization, they can be considered a spatially aware noise filtration technique [15]. A wavelet transform decomposes a signal into coefficients at multiple scales; in our case, we obtain coefficients at 4Hz, 2Hz, and 1Hz. Because the Haar wavelet transform involves computing the degree of relatedness between subsequent points in the original signal, it is excellent for detecting edges and sharp changes [15]. Using this technique applied to the participant's full EDA signal, the 3 levels of detail coefficients were computed, and statistics were computed on the coefficients over each 5-second epoch.

D. Feature Selection

Because we computed a large number of potentially redundant features, we used wrapper feature selection to ensure that our classifier did not overfit the training data. Unlike simple filtering techniques that merely rank features based on their relationship to the classification label, Wrapper feature selection (WFS) repeatedly tests subsets of features using a specific classifier to select an independent subset of features that work well in combination with each other [4]. Since this is computationally expensive, we used a greedy search process, which can quickly search the space of all subsets and is robust to overfitting [4].

E. Classification

In order to perform feature and model selection, we partitioned the data set into training, validation, and testing sets, using a randomized 60/20/20% split. Feature selection was performed using only the training data. In order to find a suitable machine learning technique for this problem, we tested a variety of algorithms including neural networks, random forests, naïve Bayes, nearest neighbour, logistic regression, and support vector machines (SVM). The algorithm that produced the best accuracy on the validation data set was SVM, so we focus on SVM for the remainder of the paper. In order to perform model selection we tested a range of settings for the parameters of SVM, including both a Radial Basis Function (RBF), polynomial, and linear kernel, and selected the settings that produced the highest accuracy on the validation set. The held-out test set was not used in feature or model selection.

IV. RESULTS

A. Classification results

Table III shows the classification results obtained for both the binary and multiclass classifiers on the validation and test sets, as well as the optimal SVM parameters. Although the accuracy for the multiclass classifier is lower (three-class classification is a more difficult problem), the output may prove more useful for real users. Fig. 3 shows both algorithms applied the same portion of EDA signal. As is evident from the figure, portions of the signal containing artifacts are detected (in red), while normal SCRs are labeled clean. Fig. 4 shows the performance of the algorithms on another sample containing a greater number of artifacts, which are also detected by both algorithms. The multiclass algorithm is able to label questionable parts of the data that are not clear artifacts in grey. Note that the binary classifier labels some epochs as artifacts that the multiclass one does not. The level of stringency needed in the classifier may depend on the researchers' application; computing aggregate measures like area under the curve may be less sensitive to artifacts than SCR detection.

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Parameter settings</th>
<th>Baseline Accuracy</th>
<th>Validation Accuracy</th>
<th>Test Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Binary</td>
<td>RBF, $\beta=0.1$, $C=1000$</td>
<td>76.0%</td>
<td>96.95%</td>
<td>95.67%</td>
</tr>
<tr>
<td>Multiclass</td>
<td>RBF, $\beta=0.1$, $C=1000$</td>
<td>61.13%</td>
<td>88.38%</td>
<td>78.93%</td>
</tr>
</tbody>
</table>

B. Features selected

The feature selection process only led to a marginal improvement in classification on the validation set: 1.3% and 1.4% for the binary and multiclass classifiers, respectively. However the features selected provide valuable insight into the signal characteristics that best distinguish between normal EDA and an artifact. Table IV shows the features selected by the binary classifier; the multiclass version selected extremely similar features. The selected features confirm the theoretical assumption that shape, including first and second derivative, are important in detecting artifacts. The wavelet features also proved valuable, especially the standard deviation of the coefficients. This is intuitive, because these values indicate whether there is a change in the wavelet domain, which may be indicative of an edge or sharp change in the original signal.
**TABLE IV**

<table>
<thead>
<tr>
<th>Category</th>
<th>Specific Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Raw SC</td>
<td>amplitude: mean</td>
</tr>
<tr>
<td></td>
<td>1st derivative: max absolute value</td>
</tr>
<tr>
<td></td>
<td>2nd derivative: max, mean absolute value</td>
</tr>
<tr>
<td>Filtered SC</td>
<td>amplitude: mean</td>
</tr>
<tr>
<td></td>
<td>2nd derivative: min, max absolute value</td>
</tr>
<tr>
<td>Wavelet</td>
<td>Mean: 1st coefficient</td>
</tr>
<tr>
<td></td>
<td>St. Dev: 1st, 2nd, 3rd coefficients</td>
</tr>
<tr>
<td></td>
<td>Median: 3rd coefficient</td>
</tr>
</tbody>
</table>

V. CONCLUSION

In summary, we have developed algorithms that can automatically and accurately distinguish artifacts in an EDA signal from normal physiological responses. The code we have written to develop these algorithms is freely available on our website, and we are currently extending the site so that anyone will be able to upload their raw EDA signal and receive an output indicating which portions contain noise. This tool could be enormously time-saving to researchers dealing with large data sets involving many participants measured over long periods of time. In the future we hope to extend our approach using active, semi-supervised learning, which will allow the machine learning algorithm to interactively ask the user to label specific epochs based on its level of uncertainty. This way, human raters will be required to label fewer epochs that are highly similar, and instead will only label novel data for which the classifier has little information.

ACKNOWLEDGMENT

This work was supported by the MIT Media Lab Consortium, Samsung, NIH Grant R01GM105018, Canada’s NSERC program, and the People Programme (Marie Curie Actions) of the European Union’s Seventh Framework Programme FP7/2007-2013/ under REA grant agreement #327702.

REFERENCES

Appendix B

Active Learning for Electrodermal Activity Classification

This paper was published in the proceedings of the IEEE Signal Processing in Medicine and Biology Symposium (SPMB) which was held in December, 2015 [45].
Active Learning for Electrodermal Activity Classification

Victoria Xia¹, Natasha Jaques¹, Sara Taylor¹, Szymon Fedor¹, and Rosalind Picard¹

Abstract—To filter noise or detect features within physiological signals, it is often effective to encode expert knowledge into a model such as a machine learning classifier. However, training such a model can require much effort on the part of the researcher; this often takes the form of manually labeling portions of signal needed to represent the concept being trained. Active learning is a technique for reducing human effort by developing a classifier that can intelligently select the most relevant data samples and ask for labels for only those samples, in an iterative process. In this paper we demonstrate that active learning can reduce the labeling effort required of researchers by as much as 84% for our application, while offering equivalent or even slightly improved machine learning performance.

I. INTRODUCTION

In order to extract meaningful information from physiological signals, researchers are often forced to painstakingly review large quantities of signal data in order to determine which portions contain poor quality signal or noise, and which contain useful information. In the case of large-scale studies which gather hundreds of thousands of hours of noisy, ambulatory data (e.g., [22]), this is extremely impractical. For this reason, recent research efforts have focused on training automated algorithms to recognize noise in physiological signals, including electrocardiogram (ECG) (e.g., [16], [17], [27]), electrodermal activity (EDA) (e.g., [28]), and electroencephalography (EEG) (e.g., [30]). While these automated techniques have proven successful, training them still requires a large amount of effort on the part of human researchers.

This research focuses on more efficiently training automatic algorithms to extract meaningful information from electrodermal activity (EDA) data. EDA refers to electrical activity on the surface of the skin, which increases in response to exertion, temperature, stress, or strong emotion [2]. For this reason, EDA has been frequently used to study emotion and stress (e.g., [12], [13], [23]). Improvements in the devices used to monitor EDA have allowed for 24-hour-a-day, ambulatory monitoring, enabling important research into how the body responds psychologically to stress and emotional stimuli during daily life [22]. However, this type of in-situ monitoring comes at a price; ambulatory EDA is often too voluminous to be inspected by hand, and noisy, containing artifacts generated from fluctuations in contact or pressure between the sensor and the skin. In order to extract meaningful information from this data, it is important to automatically distinguish between skin conductance responses (SCRs) (which may indicate increased stress or emotion), and noise (see Figures 1 and 2).

Fig. 1. Raw EDA signal containing normal skin conductance responses (SCRs) that occur in response to temperature, exertion, stress, or emotion.

This figure shows a typical artifact, where the left side of the figure is the raw signal, and the right side is the signal after applying a low-pass Butterworth filter (order= 6, f₀ = 1 Hz). Simple filtering and smoothing is insufficient to remove artifacts.

Typical methods for removing noise from EDA signal involve exponential smoothing (e.g., [9]) or low-pass filtering (e.g., [10], [13], [20], [23]); these techniques are unable to deal with large-scale artifacts such as those shown in Figure 2, and may result in these artifacts being mistaken for SCRs. Similarly, commonly employed heuristic techniques for detecting SCRs (e.g., [1], [26], [24]) are also error prone. A more effective approach is to encode the knowledge of human experts into a mathematical model or machine learning algorithm (e.g., [5], [18], [28]). This approach was successfully used to train a machine learning classifier that could identify artifacts in an EDA signal with over 95% accuracy [28]. However, encoding this knowledge required significant effort; two experts had to label over 1500 five-second epochs of EDA data to achieve this recognition rate [28]. Further, this type of encoding can lead to a highly specific model that cannot generalize to other applications.
Active learning is a promising technique for encoding expert knowledge in a machine learning classifier with minimal human effort. Rather than require an expert researcher to label a dataset of thousands of examples, an active learning classifier can intelligently select which samples would be most informative to the classification problem, and only request labels for those. This paper will explore how to employ active learning in two different problems within the domain of EDA signal processing: artifact detection and detection of SCRs. Both problems showed promising results.

II. BACKGROUND

The following sections will introduce concepts and previous work related to the machine learning and active learning algorithms we use in this research.

A. Support Vector Machines

Support Vector Machines (SVMs) are a machine learning classification algorithm that have been found to be highly effective with active learning [14] [25] [29]. SVMs work by finding a decision boundary that separates one class from another (e.g., SCR vs. non-SCR). For a separable dataset, there are many possible decision boundaries that could divide the data into the two classes. SVMs choose the boundary that maximizes the size of the margin, the space between the decision boundary hyperplane and the points closest\(^1\) to it (see Figure 3 for an illustration of this concept) [8].

B. Active Learning

The purpose of active learning is to intelligently select data samples that can be used to train an effective machine learning classifier, so not all samples have to be labeled by human experts. In pool-based active learning, the active learner selects batches of data samples, or query sets, for which to request labels, from a provided collection of unlabeled samples.

A reasonable way to obtain the initial batch is to cluster the data, and include the samples closest to the centroid of each cluster in the first query set [11]. This approach allows the classifier to begin developing a representative decision boundary; increasing the number of clusters improves the initial decision boundary, but requires the human expert to provide more labels. Various strategies can then be used to select successive query sets. We will introduce several below based on active learning with SVMs. After each query set, the SVM is re-trained, and the new decision boundary produced is used to help evaluate which data samples should be included in the next query set.

The following are various SVM-based query strategies:

1) Simple Margin: This strategy is one of many uncertainty sampling algorithms. Uncertainty sampling techniques choose to query samples the classifier is unsure how to label, with the expectation that learning how to label these samples will improve the classifier the most [15]. In the case of SVMs, one heuristic for how uncertain the classifier is about each sample is the distance of the sample to the decision boundary. Thus, Simple Margin queries the samples closest to the decision boundary of the SVM [3].

2) Mirroshandel’s Algorithm (2011): As shown in Figure 3, a drawback of Simple Margin is that it may query outliers. To prevent this, Mirroshandel et al. [19] proposed balancing uncertainty with representativeness, by combining distance to the decision boundary (uncertainty) with average distance to the other samples (representativeness). These factors are combined by placing a weight of \(\alpha\) on uncertainty and \(1 - \alpha\) on representativeness.

3) Xu’s Algorithm (2003): Naively querying the samples closest to the decision boundary (as Simple Margin does) is problematic, both because these samples may not be representative of the other samples (the motivation for Mirroshandel’s algorithm), but also because these samples may be redundant among themselves. To remedy both these issues, Xu et al. [32] proposed running a clustering algorithm on the samples near the decision boundary, and then querying the centers of each cluster. Samples chosen using this method will be close to the boundary, likely representative of the surrounding samples, and different from each other. Xu’s algorithm chooses to cluster on samples within the margin.

4) MaxMin Margin: Another uncertainty sampling technique, MaxMin Margin seeks to more intelligently shrink the space of consistent decision boundaries (i.e., version space) by maximizing the guaranteed change in version space size with each query. To do so, MaxMin Margin follows the algorithm below [29]:

1) For each unlabeled sample, \(i\):
   a) Treat \(i\) as a positive example. Compute \(m^+_i\), the size of the resulting margin.
   b) Treat \(i\) as a negative example. Compute \(m^-_i\), the size of the resulting margin.
   c) Find the minimum of \(m^+_i\) and \(m^-_i\). Call it \(m_i\).
2) Query the samples that have the greatest \(m_i\) values.
3) Repeat 1) and 2) until some stopping criterion is reached.

\(^1\)Here “closest” is a measure of distance defined by the feature space and kernel function.
C. Automatic Stopping Criterion

Because the goal of active learning is to allow experts to label only a fraction of the data required for supervised learning, an important question is how to decide when enough samples have been queried. One option would be to allow human experts to periodically evaluate the performance of the classifier and stop when satisfied, but this is unreliable and time-consuming. Ideally, the active learner itself would be able to suggest a stopping point. Schohn and Cohn [25] proposed a simple stopping criterion that has been empirically proven to be highly effective: continue querying until all samples within the margin of the SVM have been labeled.

III. DATA COLLECTION

A. Dataset I

Building on previous work [28], we began by applying active learning to the problem of artifact detection in EDA. We use the same dataset that was analyzed in the original work [28], which was obtained from a study in which participants wore Affectiva Q EDA sensors while experiencing physical, cognitive and emotional stressors [6]. The collected data were split into non-overlapping five-second epochs; 1560 epochs were selected for expert labeling. Two experts labeled each epoch as either containing an artifact or not based on an agreed-upon set of criteria, using the procedure described in Section IV-A.

B. Dataset II

In order to provide a more robust test of the effectiveness of active learning, we obtained novel data collected in a different setting, for a different application, using a different sensor—the Empatica E4. The study collected data from participants while they were sleeping at home. A total of 3001 non-overlapping epochs were labeled by two experts using the procedure described in Section IV-A, but this time epochs were labeled for skin conductance responses (SCRs) rather than for artifacts. The experts agreed beforehand to label SCRs based on Boucsein’s characterization of a typical SCR [2]. Automatically detecting SCRs during sleep could potentially be very useful, especially in studying EDA “sleep storms,” bursts of activity in which many SCRs occur during slow-wave sleep [21].

IV. METHODS

A. Expert Labeling

Our experts labeled each epoch as belonging to one of two classes (artifact or non-artifact for Dataset I, SCR or non-SCR for Dataset II) using an online tool we built, EDA Explorer (eda-explorer.media.mit.edu). The site automatically splits raw EDA data into five-second epochs, generates and displays plots for epochs one at a time, and records input labels associated with each plot. For each epoch, the experts were shown a plot of both the raw and low-pass-filtered five-second EDA signal, a context plot showing the surrounding five seconds of signal on either side (if applicable), and also plots of the relevant accelerometer and temperature signals (collected by the sensor). Using these plots, each expert chose to assign one of two possible labels to the epoch, or skip it (if they did not wish to assign a label).

Note that although the experts were shown accelerometer and temperature plots to aid in their decision-making, this information was not provided to any of our classifiers, as in the original work [28]. By withholding this information from our classifiers, we allow them to generalize to EDA signal collected by any EDA sensor, regardless of whether accelerometer and temperature data were also collected.

B. Partitioning Data

For each dataset, epochs which were skipped and those for which the two experts did not agree on a label were removed, following the original work [28]. This choice was made because it is impossible to determine a ground truth classification for epochs on which the experts disagree. Further, we seek to establish the effectiveness of active learning in comparison to previous work on automatic signal classification, and using the same dataset as previous work [28] allows us to do so. For Dataset II, the experts identified four times as many epochs that did not contain an SCR as epochs that did, so we randomly subsampled non-SCR epochs to create a balanced dataset.

In total, Dataset I contained 1050 epochs, and Dataset II contained 1162. The epochs for each dataset were randomly split into training, validation, and test sets in a 60/20/20% ratio. Feature selection was performed using the training data, parameter selection was performed using the validation data, and the testing data was held-out until the algorithms were finalized, and used to provide an estimate of the classifier’s generalization performance.

C. Feature Extraction

From each dataset, we extracted the same features as in the original work [28]. These features included shape features (e.g., amplitude, first and second derivatives) of both the raw signal and the signal after a 1Hz low-pass filter had been applied. We computed additional features by applying a Discrete Haar Wavelet Transform to the signal. Wavelet transforms are a time-frequency transformation; the Haar wavelet transform computes the degree of relatedness between subsequent samples in the original signal, and therefore can detect edges and sharp changes [31]. We obtained wavelet coefficients at 4Hz, 2Hz, and 1Hz, and computed statistics related to those coefficients for each five-second epoch. At this stage we were unconcerned with redundancy of features, as we later performed feature selection (see Section IV-D). Finally, we standardized each feature to have mean 0 and variance 1 over epochs in the training set.

D. Feature Selection

The features provided to the classification algorithm were selected using wrapper feature selection (WFS) applied to
Dataset I, as in the original work [28]. WFS tests classifier performance using different subsets of features, and aims to select the subset of features which yields the best performance [7]. Because this is computationally expensive, we used a greedy search strategy to search through the space of possible subsets. Because feature selection requires knowledge of the true classification label, we did not re-select new features for Dataset II. We reasoned this would be a more robust test of active learning, in which the true classification label of the data samples in the training set is not known in advance, and allow us to assess how our active learning pipeline will generalize to novel EDA data.

E. Active Learning

The previous work on detecting artifacts in an EDA signal explored many different machine learning algorithms [28], and found SVMs to be the most effective. For this reason, and because SVMs make for excellent active learners, we chose to focus on the SVM classifier for this work.

Our active learning methodology is as follows. The initial query set provided to the active learner was determined by running an approximate k-means algorithm to cluster the unlabeled samples into \texttt{num-clusters} clusters, and choosing the sample closest to the centroid of each cluster to be part of the query set\(^2\). Subsequent query sets of size \texttt{batch-size} samples were selected using one of four different query strategies. After assessing the effectiveness of a range of values, we selected \texttt{num-clusters} = 30 and \texttt{batch-size} = 5.

The four query strategies we tested were discussed above (in II-B): Simple Margin, Mirroshandel’s representativeness algorithm, Xu’s clustering algorithm, and MaxMin Margin. For our implementation of Mirroshandel’s algorithm, we tested \(\alpha\) values ranging from 0.40 to 0.75, based on values reported in previous work [19]. For Xu’s algorithm, we experimented with clustering on varying amounts of samples closest to the boundary (e.g., closest 5\% of unlabeled samples to the boundary, closest 30 unlabeled samples, etc.).

In order to assess whether our active learning methods provide a benefit over a simple passive learner, we also implemented a query strategy which simply chooses the query set randomly. Note that this random sampler still benefits from the initial query set provided by the clustering algorithm. Any improvements upon the random sampler must therefore be due to the benefit of active learning, rather than the clustering.

We also assessed the effectiveness of employing the stopping criterion presented in Section II-C to the Simple Margin query strategy. We did not test it with the other techniques; the criterion does not apply well to them, since they are not are not guaranteed to query samples within the margin first.

V. RESULTS

Because the approximate clustering algorithm used to select the initial query set contains randomness, query sets (and consequently, classifier performance) vary across different runs of the algorithm. Thus, all of our reported results are averaged over 100 runs. We report our results in terms of accuracy; that is, the proportion of correctly classified examples in the validation or test set.

A. Dataset I - Artifacts

We began by searching the parameter space of each active learning algorithm to find the values that gave the best performance on the validation set. For Xu’s clustering algorithm, we found clustering on the 30 samples closest to the margin to be most effective. For the representativeness algorithm, \(\alpha = 0.65\) was found to give the best performance, which agrees with what was found previously [19].

Figure 4(a) shows the performance of each strategy in classifying artifacts in Dataset I. We see that three of the techniques outperform the Random baseline, indicating that active learning on this problem is highly effective. Using Simple Margin, for example, a higher validation accuracy is obtained after only 100 samples have been labeled than is ever obtained by passive learning. After this initially high performance, the accuracy of Simple Margin drops steadily to meet the level of the classifier trained on the entire dataset. Others have observed this phenomenon as well (e.g., [4], [25]). We suspect the initially superior performance of the active learning methods may be due to their ability to successfully query relevant samples. As the rest of the data, including irrelevant examples, are added to the classifier, the classifiers may begin to overfit to the training data [4].

In comparing the performance of Simple Margin with that of the Xu’s and Mirroshandel’s algorithm, we see that they are roughly equivalent for the first 300 samples. After this point, Mirroshandel’s and Xu’s algorithm appear to outperform Simple Margin. Because the goal of active learning is to have the human researcher label as few examples as possible, this later performance difference is of little interest. However, we can see from Figure 4(a) that MaxMin Margin does not offer a performance improvement over Random Sampling. We suspect this is because our dataset may not be separable, due to a combination of noise and having only selected nine features, thus violating the separability assumption of MaxMin Margin [29].

Figure 4(b) and Table I show the results obtained using Schohn and Cohn’s automatic stopping criterion. On average, the algorithm chose to stop after only 96.7 samples had been queried, and ended with an average validation accuracy greater than that achieved using the entire training set and reported in the original work [28]. Because this stopping criterion is so effective, we chose to apply Simple Margin with this criterion to the testing data to obtain a final estimate of the ability of our technique to generalize to novel data. Figure 4(c) shows the accuracy obtained on the held-out test dataset. As with the validation set, Simple Margin with Schohn and Cohn’s automatic stopping criterion ended, on average, with a test accuracy higher than that achieved by training on the entire training set. This suggests that with active learning, the same (or even slightly better)
classification performance can be achieved with one-sixth the number of labeled training epochs, allowing for a large reduction in the amount of labor required of human experts.

B. Dataset II - Peaks

To demonstrate that active learning is also effective for novel problems and novel sensor technologies, we apply the same methodology to Dataset II in order to detect SCRs within an EDA signal. In this case, we found clustering on the 5% of peaks closest to the decision boundary to be most effective with Xu’s algorithm, and once again found $\alpha = 0.65$ to work best with Mirroshandel’s algorithm.

As shown in Figure 5(a), we observed similar relative performance of the various query strategies for this second dataset as for the first. Note that the performance advantage offered by additional training data is smaller. We suspect this is because the problem is easier; SCRs tend to have a consistent shape whereas artifacts and noise can vary. Schohn and Cohn’s automatic stopping criterion performed well once again (see Figures 5(b) and Table II). On average, the algorithm stopped after 155.4 samples (23% of the entire training set), and still achieved slightly better performance than training on the entire dataset. In Figure 5(c), we plot the performance of the active learner on the held-out test dataset. We can see once again that the generalization performance of active learning is higher than that of passive learning, yet requires only 23% of the effort on the part of human researchers.

VI. CONCLUSION AND FUTURE WORK

We have shown that active learning is a promising technique for reducing the labeling effort required to train a machine learning algorithm in signal classification. In our case, the active learner can achieve equivalent or even slightly superior performance using as little as 16% of the data. We found Simple Margin to be an effective and reliable strategy, which if deployed with Schohn and Cohn’s stopping criteria can automatically determine when to stop querying for more labels. The results from Dataset II are of particular interest because they correspond to the performance we can expect when employing our active learning techniques on a novel EDA dataset.

Though our exploration focused on EDA signals, our techniques could likely be extended to other types of signals, such as ECG or photoplethysmogram (PPG). We are also planning to incorporate our active learning algorithm into the online tool we have built for processing EDA signals, eda-explorer.media.mit.edu. Researchers will be able to use our site to label data to train a classifier, while the active learner intelligently selects which epochs need to be labeled. Future work may also include extending the active learning techniques for EDA signal classification discussed here to multiclass classification. For example, we may be interested in classifying epochs into one of three categories: artifact, SCR, or neither.
Fig. 5. (a) Validation accuracies of various query strategies tested on the peaks dataset; (b) Validation performance of Schohn and Cohn’s automatic stopping criterion on the peaks dataset. The large red dot marks the average number of samples queried before the algorithm chose to stop, and the average ending validation accuracy; (c) Generalization performance (test accuracy) of Simple Margin with Schohn and Cohn’s automatic stopping criterion, on the peaks dataset. The large red dot marks the average number of samples queried and the average ending test accuracy.

ACKNOWLEDGMENT

This work was supported by the MIT Media Lab Consortium, the Robert Wood Johnson Foundation, Canada’s NSERC program, and the People Programme of the European Union’s Seventh Framework Programme.

REFERENCES

Appendix C

State-Space Generalized Linear Model Derivation

In Chapter 5, we discussed the State-Space Generalized Linear Model (SS-GLM) presented by Czanner et. al. [13] and how we can model electrodermal responses (EDRs) using the model. In this appendix, we will give the derivations of the Expectation-Maximization equations used to estimate the parameters of the model.

C.1 Model Set-up

First, we define the variables used in the SS-GLM. Then we re-state the observation and state equation of the model for reference (these were first introduced in Section 5.2).

C.1.1 Definitions of Variables

- Total number of nights: $K$
- Night index: $k \in \{1, \ldots, K\}$
- Total duration of the night (in seconds): $T$
- Number of sub-intervals each night is divided into: $L$
- Sub-interval width (in seconds): $\Delta \equiv TL^{-1}$
- Sub-interval index: $l \in \{1, \ldots, L\}$
- Binary EDR event indicator at night $k$ and time $l\Delta$: $n_{k,l}$
- Full EDR event dataset: $N$
- Total number of time since sleep onset functions: $R$
- Function index: $r \in \{1, \ldots, R\}$
- Time since sleep onset effect: $\theta_k \in \mathbb{R}^R$
- Time since sleep onset functions: $g_r(l\Delta)$
- Time since sleep onset rate (in EDRs/second): $\lambda^S(l\Delta|\theta_k)$
- Total number of recent history windows: $J$
- History index: $j \in \{1, \ldots, J\}$
- History of events in night $k$ up to, but not including $l\Delta$: $H_{k,l} = \{n_{k,l-J}, \ldots, n_{k,l-2}, n_{k,l-1}\}$
- History effect: $\gamma \in \mathbb{R}^J$
- History rate (dimensionless): $\lambda^H(l\Delta|\gamma, H_{k,l})$
- Rate of EDR events (in EDRs/second): $\lambda_k(l\Delta|\theta_k, \gamma, H_{k,l})$
- Initial time since sleep onset effect: $\theta_0$
- State covariance matrix: $\Sigma$ of size $R \times R$

### C.1.2 Observation Equation - Conditional Intensity Function (CIF)

The rate of EDR events is a combination of the **time since sleep onset effect** ($\lambda^S$, units of peaks/second) which models the effect of the time since sleep onset on EDRs and **history effect** ($\lambda^H$, dimensionless) which models the effect of the history of EDR events on current activity$^1$

\[
\lambda_k(l\Delta|\theta_k, \gamma, H_{k,l}) = \lambda^S(l\Delta|\theta_k)\lambda^H(l\Delta|\gamma, H_{k,l}) \tag{C.1}
\]

\[
\log \left[ \lambda^S(l\Delta|\theta_k) \right] = \sum_{r=1}^R \theta_{k,r} g_r(l\Delta) \tag{C.2}
\]

\[
\log \left[ \lambda^H(l\Delta|\gamma, H_{k,l}) \right] = \sum_{j=1}^J \gamma_j n_{k,l-j} \tag{C.3}
\]

$^1$The history effect is the same across nights and therefore seeks to describe the underlying physiological process
C.1.3 State Equation

\[ \theta_k = \theta_{k-1} + \epsilon_k \]  

(C.4)

where \( \epsilon_k \) is distributed normally with mean 0 and covariance \( \Sigma \). We note that \( \Sigma \) is an \( R \) dimensional Gaussian random vector and we assume \( \theta_0 \) is unknown.

C.2 Expectation-Maximization

From the previous section, we end up with several unknown parameters \( \psi = (\gamma, \theta_0, \Sigma) \) and an unobservable variable \( \theta_k \) that we need in order to estimate the rate of EDR events. In models, like the SS-GLM, where we wish to estimate latent variables without computing the negative log likelihood, the expectation-maximization (EM) algorithm is often used [31]. The EM algorithm alternates between inferring the unobservable variables given the parameters (E-step) and optimizing the parameters given the unobservable variables (M-step) until the estimates have converged.

In particular, we define the auxiliary function, \( Q \), to be the expected complete data log likelihood:

\[ Q(\psi|\psi^{(i)}) = \mathbb{E}[l_c(\psi)|N, \psi^{(i)}] \]

where \( i \) is the iteration number and \( l_c \) is the complete data log likelihood

\[ l_c(\psi) \equiv \log p(\theta, N|\psi) \]

During the E-step, we compute the terms used to compute \( Q(\psi|\psi^{(i)}) \). Then, during the M-step we optimize the \( Q \) function with respect to \( \psi \) to get a new estimate for \( \psi \):

\[ \psi^{(i+1)} = \arg \max_{\psi} Q(\psi|\psi^{(i)}) \]

In the next sections, we derive the EM update equations for the SS-GLM.

C.2.1 Complete Data Likelihood

We let \( \lambda_k = \lambda_k(l\Delta|\theta_k, \gamma, H_{k,l}) \) for brevity.

First we note that the complete data likelihood can be written as:

\[ p(\theta, N|\psi) = p(N|\theta, \psi) \times p(\theta|\psi) \]  

(C.5)

Because our point process is characterized by a conditional density function, we can write \( p(N|\theta, \psi) \) as a joint probability mass function:

\[ p(N|\theta, \psi) = \exp \left\{ \sum_{k=1}^{K} \sum_{l=1}^{L} [n_{k,l} \log (\lambda_k \Delta) - \lambda_k \Delta] \right\} \]  

(C.6)
Furthermore, since the state process is a Gaussian random walk model, we can write $p(\theta|\psi)$ as a joint probability density:

$$p(\theta|\psi) = \left[ (2\pi)^{-R/2} |\Sigma|^{-1/2} \right]^K \times \exp \left\{ -\frac{1}{2} \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right\} \quad (C.7)$$

### C.2.2 E-Step

Recall that the goal of the E-step is to compute the terms used to compute $Q(\psi|\psi^{(i)})$. Therefore, we first split the expectation into two parts since $\log(A \cdot B) = \log(A) + \log(B)$ and expectation operator, $E[\cdot]$, is a linear operator:

$$Q(\psi|\psi^{(i)}) = E[\log p(\theta, N|\psi^{(i)})] = E[\log p(N|\theta, \psi^{(i)})] + E[\log p(\theta|\psi^{(i)})] \quad (C.8)$$

Evaluating the first component of $Q(\psi|\psi^{(i)})$,

$$E[\log p(N|\theta, \psi^{(i)})] = E \left\{ \sum_{k=1}^{K} \sum_{l=1}^{L} \left[ \log (\lambda_k \Delta) - \lambda_k \Delta \right] \right\}_{N, \psi^{(i)}}$$

$$= \sum_{k=1}^{K} \sum_{l=1}^{L} \left( E \left[ \log (\lambda_k \Delta) \right] - E \left[ \lambda_k \Delta \right] \right) \quad E \text{ is linear}$$

$$= \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} E \left[ \log (\lambda_k \Delta) \right] - E \left[ \lambda_k \Delta \right] \right) \quad n_{k,l} \text{ is known}$$

Using Eqn. [C.1] we compute $E \left[ \log (\lambda_k \Delta) \right]$:

$$E \left[ \log (\lambda_k \Delta) \right] = E \left[ \log \Delta + \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) + \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right] \quad \Delta \text{ is constant}$$

$$= \log \Delta + \sum_{r=1}^{R} E \left[ \theta_{k,r} g_r(l\Delta) \right] + \sum_{j=1}^{J} \gamma_j n_{k,l-j} \quad \gamma, N \text{ are known}$$

$$= \log \Delta + \sum_{r=1}^{R} E \left[ \theta_{k,r} \right] g_r(l\Delta) + \sum_{j=1}^{J} \gamma_j n_{k,l-j} \quad g_r(l\Delta) \text{ is independent of } \theta_{k,r}$$

We also compute $E \left[ \lambda_k \Delta \right]$: 
\[ E[\lambda_k \Delta] = E \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l \Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \Delta \right] \]

\[ = E \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l \Delta) \right\} \right] \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \Delta \]

Putting it all together we have the following for the first component of \( Q(\psi|\psi^{(i)}) \):

\[ E[\log p(N|\theta, \psi^{(i)})] = \sum_{k=1}^{K} \sum_{l=1}^{L} n_{k,l} \left( \log \Delta + \sum_{r=1}^{R} E[\theta_{k,r}] g_r(l \Delta) + \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right) \]

\[ - E \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l \Delta) \right\} \right] \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \Delta \]

Then evaluating the second component of \( Q(\psi|\psi^{(i)}) \), we have:

\[ E[\log p(\theta|\psi^{(i)})] = - \frac{KR}{2} \log(2\pi) - \frac{K}{2} \log |\Sigma| - \frac{1}{2} E \left[ \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right| N, \psi^{(i)} \] (C.9)

We then evaluate\(^2\) the last term in Eqn. C.9.

\(^2\)Special thanks to Ben Ehlert for helping me figure this out.
\[
\mathbb{E} \left[ \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right]
\]
\[
= \sum_{k=1}^{K} \mathbb{E} \left[ (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right]
\]
\[
= \sum_{k=1}^{K} \mathbb{E} \left[ \text{tr} \left( (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right) \right]
\]
\[
= \mathbb{E} \left[ \text{tr} \left( \Sigma^{-1} (\theta_k - \theta_{k-1}) (\theta_k - \theta_{k-1})^T \right) \right]
\]
\[
= \text{tr}(A + B) = \text{tr}(A) + \text{tr}(B)
\]
\[
= \text{tr} \left( \sum_{k=1}^{K} \Sigma^{-1} (\theta_k - \theta_{k-1}) (\theta_k - \theta_{k-1})^T \right)
\]
\[
\Sigma \text{ is known}
\]
\[
= \text{tr} \left( \sum_{k=1}^{K} \Sigma^{-1} \left[ \mathbb{E} \left[ \theta_k \theta_k^T \right] - 2 \mathbb{E} \left[ \theta_k \theta_{k-1}^T \right] + \mathbb{E} \left[ \theta_{k-1} \theta_{k-1}^T \right] \right] \right)
\]
\[
\text{multiply out, } \mathbb{E}[\cdot] \text{ is linear}
\]
\[
= \text{tr} \left( \sum_{k=1}^{K} \Sigma^{-1} \left[ \mathbb{E} \left[ \theta_k \theta_k^T \right] - 2 \mathbb{E} \left[ \theta_k \right] \mathbb{E} \left[ \theta_{k-1} \right]^T \right.ight.
\]
\[
\left. -2 \text{cov}(\theta_k, \theta_{k-1}) + \mathbb{E} \left[ \theta_{k-1} \theta_{k-1}^T \right] \right] \right)
\]
\[
\mathbb{E}[XY] = \mathbb{E}[X] \mathbb{E}[Y] + \text{cov}(X, Y)
\]

Thus, in order to evaluate \(Q(\psi | \psi^{(i)})\) we need to be able to evaluate the following four expressions (the first two are from the first term of \(Q\), and the second two from the second term of \(Q\)):

1. \(\theta_{k|K} \equiv \mathbb{E} \left[ \theta_k \mid N, \psi^{(i)} \right]\)

2. \(\mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l \Delta) \right\} \mid N, \psi^{(i)} \right]\)

3. \(W_{k,k+1|K} \equiv \text{cov} \left( \theta_k, \theta_{k+1} \mid N, \psi^{(i)} \right)\)
4. $\mathbb{E} \left[ \theta_k \theta_k^T \mid N, \psi^{(i)} \right]$

We use the notation $k|k'$ to denote the expectation of the unobserved process for night $k$ given $N_{1:k'}$ (the EDR event activity from night 1 to night $k'$).

We compute these terms in 3 steps using a recursive filtering algorithm, a fixed interval smoothing algorithm, and a state-space covariance algorithm.

**E-step I**

First, we note that because the state evolves as a Gaussian random walk, that the one-step prediction distribution is Gaussian. We can derive a recursive expression for the posterior mean $\theta_k|k$ and variance $W_{k|k}$ of the state using previously estimated means and variances of the state and observed EDR events (see Eden et. al. for more information [16]). Let $\theta_{k|k-1}$ and $W_{k|k-1}$ be the mean and variance of the one-step prediction density. The generic recursive linear filtering algorithm is detailed in Algorithm 1.

**Algorithm 1 Generic Linear Filtering Algorithm**

Set $\theta_{1|0}$
Set $W_{0|0}$

for $k = 1, 2, \ldots, K$ do

Let $\lambda_k = \lambda_k(l\Delta|\theta_k, \gamma, H_{k,l})$ and evaluate gradients at $\theta = \theta_{k|k-1}$

$\theta_{k|k-1} = \theta_{k-1|k-1}$

$W_{k|k-1} = W_{k-1|k-1} + \Sigma$

$W_{k|k} = \left[ -W_{k|k-1}^{-1} + \sum_{l=1}^{L} \left( \nabla^2 \log \lambda_k \times [n_{k,l} - \lambda_k \Delta] - \nabla \log \lambda_k \cdot \nabla \lambda_k \cdot \Delta \right) \right]^{-1}$

$\theta_{k|k} = \left[ \theta_{k|k-1} + W_{k|k} \sum_{l=1}^{L} \left( \lambda_k^{-1} \times \frac{\delta \lambda_k}{\delta \theta} [n_{k,l} - \lambda_k \Delta] \right) \right]$ end for

We recall that the state process begins with a constant vector, $\theta_0$. Therefore, $\theta_{1|0} = \theta_0$ and $W_{0|0} = 0$. Then we proceed forwards from $k = 1$ until $k = K$.

In our use of this model, we use unit-step functions that are the same size and non-overlapping. We can then compute the following gradients with respect to $\theta_{k|k-1}$:

$$\nabla \log \lambda_k = \nabla \left( \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) + \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right)$$

$$= \nabla \left( \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right)$$

$$= [g_1(l\Delta), \ldots, g_R(l\Delta)]$$
Therefore,
\[ \nabla^2 \log \lambda_k = 0_{R \times R} \]

Furthermore,
\[
\nabla \lambda_k = \nabla \left( \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right) \\
= [g_1(l\Delta), \ldots, g_R(l\Delta)] \lambda_k
\]

We can then compute \( \nabla \log \lambda_k \cdot \nabla \lambda_k \cdot \Delta \) as follows:

\[
\nabla \log \lambda_k \cdot \nabla \lambda_k \cdot \Delta = \begin{bmatrix}
(g_1(l\Delta))^2 & g_1(l\Delta)g_2(l\Delta) & \ldots & g_1(l\Delta)g_R(l\Delta) \\
g_2(l\Delta)g_1(l\Delta) & (g_2(l\Delta))^2 & \ldots & g_2(l\Delta)g_R(l\Delta) \\
\vdots & \vdots & \ddots & \vdots \\
g_R(l\Delta)g_1(l\Delta) & g_R(l\Delta)g_2(l\Delta) & \ldots & (g_R(l\Delta))^2
\end{bmatrix} \lambda_k \Delta \\
= G(l\Delta) \lambda_k \Delta
\]

We point out that because we are using non-overlapping unit step functions for the \( g_r \) functions, \( G(l\Delta) \) simplifies to having only one entry of \( 1 \) and the rest of the entries are \( 0 \). In particular, there will be a \( 1 \) in the \( (r, r) \)th cell of \( G(l\Delta) \) when \( l\Delta \) satisfies \((r-1)\frac{T}{R} + 1 \leq l\Delta < r\frac{T}{R}\).

Therefore, for our model set up, the linear recursive filtering algorithm simplifies to Algorithm 2.

**Algorithm 2 SS-GLM using Unit-Step Functions Linear Filtering Algorithm**

Set \( \theta_{1|0} = \theta_0 \)
Set \( W_{0|0} = 0 \)
for \( k = 1, 2, \ldots, K \) do
Let \( \lambda_k = \lambda_k(l\Delta|\theta_k, \gamma, H_k,l) \)
\( \theta_{k|k-1} = \theta_{k-1|k-1} \)
\( W_{k|k-1} = W_{k-1|k-1} + \Sigma \)
\( W_{k|k} = \left[ W_{k|k-1} + \sum_{l=1}^{L} G(l\Delta) \lambda_k \Delta \right]^{-1} \)
\( \theta_{k|k} = \left[ \theta_{k|k-1} + W_{k|k} \sum_{l=1}^{L} [g_1(l\Delta), \ldots, g_R(l\Delta)]^T n_{k,l} - \lambda_k \Delta \right] \)
end for
E-step II

After getting estimates for the posterior mean and variance, we use the fixed-interval smoothing algorithm described by Mendel [30] and Smith and Brown [40] to compute $\theta_{k|K}$ and $W_{k|K}$ (see Algorithm 3).

Algorithm 3 Fixed-Interval Smoothing Algorithm

\begin{algorithm}
\begin{algorithmic}
  \FOR {$k = K - 1, \ldots, 2, 1$}
    \STATE Let $A_k = W_{k|k}W_{k+1|k}^{-1}$
    \STATE $\theta_{k|K} = \theta_{k|k} + A_k(\theta_{k+1|K} - \theta_{k+1|k})$
    \STATE $W_{k|K} = W_{k|k} + A_k(W_{k+1|K} - W_{k+1|k})A_k^T$
  \ENDFOR
\end{algorithmic}
\end{algorithm}

Thus, we have computed one of our needed expressions of $Q$:

$$\theta_{k|K} \equiv \mathbb{E} \left[ \theta_k \mid N, \psi^{(i)} \right]$$  \hfill (C.10)

We can also compute $\mathbb{E} \left[ \theta_k\theta_k^T \mid N, \psi^{(i)} \right]$ since $\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y] + \text{cov}(X,Y)$. In particular, we have

$$\mathbb{E} \left[ \theta_k\theta_k^T \mid N, \psi^{(i)} \right] = W_{k|K} + \theta_{k|K}\theta_{k|K}$$  \hfill (C.11)

E-step III

We then use the state-space covariance algorithm described by De Jong and MacKinnon [15] to compute the covariate estimate as follows:

$$W_{k,u|K} = \text{Cov} \left[ \theta_k, \theta_u \mid N, \psi^{(i)} \right] = A_k\text{Cov} \left[ \theta_{k+1}, \theta_u \mid N, \psi^{(i)} \right]$$

Thus, when we let $u = k + 1$, we can compute another needed expression of $Q$, namely $W_{k,k+1|K}$:

$$W_{k,k+1|K} = \text{Cov} \left[ \theta_k, \theta_{k+1} \mid N, \psi^{(i)} \right] = A_k\text{Cov} \left[ \theta_{k+1}, \theta_{k+1} \mid N, \psi^{(i)} \right]$$  \hfill (C.12)

Finally, we need to compute $\mathbb{E} \left[ \exp \left\{ \sum_{r=1}^R \theta_{k,r} g_r(l\Delta) \right\} \mid N, \psi^{(i)} \right]$. First we note that $\theta_{k,r}$ is normally distributed with mean $\theta_{k|K,r}$ and variance $W_{k|K,r,r}$ (where $W_{k|K,r,r}$ is the $r; r$th element of $W_{k|K}$). This can be written as:
\[
\begin{bmatrix}
\theta_{k,1} \\
\theta_{k,2} \\
\vdots \\
\theta_{k,R}
\end{bmatrix}
\sim N
\begin{pmatrix}
\begin{bmatrix}
\theta_{k|K,1} \\
\theta_{k|K,2} \\
\vdots \\
\theta_{k|K,R}
\end{bmatrix}, W_{k|K}
\end{pmatrix}
\]

Therefore, the distribution of \( \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \) is a sum of dependent Gaussians and is therefore Gaussian itself.

Let \( g(l\Delta) = [g_1(l\Delta), g_2(l\Delta), \ldots, g_R(l\Delta)] \). Then \( \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \) is normally distributed with mean \( \sum_{r=1}^{R} \theta_{k|K,r} g_r(l\Delta) \) and variance \( g(l\Delta) \cdot W_{k|K} \cdot g(l\Delta)^T \).

By the Law of the Unconscious Statistician (LUS), we know that

\[
\mathbb{E}[g(X)] = \int_{-\infty}^{\infty} g(x) f(x) dx
\]

where \( f(x) \) is the probability distribution of \( X \).

Thus, if \( X \) is normally distributed with mean \( \mu \) and variance \( \sigma^2 \), then we can define \( X = \mu + \sigma Z \) where \( Z \sim N(0, 1) \).

Therefore,

\[
\begin{align*}
\mathbb{E}[e^X] &= \mathbb{E}[e^{\mu+\sigma Z}] \\
&= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{\mu+\sigma z} e^{-z^2/2} dz \\
&= \frac{e^\mu}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{\sigma z - z^2/2} dz \\
&= \frac{e^\mu}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-(z-\mu)^2/2} dz \\
&= \frac{e^\mu}{\sqrt{2\pi}} \sqrt{2\pi} \\
&= e^\mu \\
&\approx e^\mu + e^\mu (\mu + \frac{1}{2}\sigma^2 - \mu) \\
&= e^\mu [1 + \frac{1}{2}\sigma^2]
\end{align*}
\]

Now we can use the expression for \( \mathbb{E}[e^X] \) in order to compute \( \mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \mid N, \psi^{(i)} \right] \) as follows:
\[
\mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k|K,r} g_r(l\Delta) \right\} \right] = \exp \left\{ \sum_{r=1}^{R} \theta_{k|K,r} g_r(l\Delta) \right\} \\
\cdot \left[ 1 + \frac{1}{2} \left( g(l\Delta) \cdot W_{k|K} \cdot g(l\Delta)^T \right) \right]
\]

Since we are using non-overlapping unit step functions, this equation reduces further. Let \( r \) be the integer that satisfies \( (r - 1) \frac{T}{R} + 1 \leq l\Delta < r \frac{T}{R} \). Then

\[
\exp \left\{ \sum_{r=1}^{R} \theta_{k|K,r} g_r(l\Delta) \right\} = \exp \{ \theta_{k|K,r} \}
\]

and

\[
g(l\Delta) \cdot W_{k|K} \cdot g(l\Delta)^T = W_{k|K,r,r}
\]

Therefore, using the appropriate \( r \) defined by \( l\Delta \), we have:

\[
\mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k|K,r} g_r(l\Delta) \right\} \right] = \exp \{ \theta_{k|K,r} \} \cdot \left[ 1 + \frac{1}{2} W_{k|K,r,r} \right] \quad (C.13)
\]

**Summary**

The E-step of EM seeks to compute the terms used to compute \( Q(\psi|\psi^{(i)}) \). We found that this required the computation of 4 expressions:

1. \( \theta_{k|K} \equiv \mathbb{E} \left[ \theta_k \mid N, \psi^{(i)} \right] \)

2. \( \mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \mid N, \psi^{(i)} \right] \)

3. \( W_{k,k+1|K} \equiv \text{cov} \left( \theta_k, \theta_{k+1} \mid N, \psi^{(i)} \right) \)

4. \( \mathbb{E} \left[ \theta_k \theta_k^T \mid N, \psi^{(i)} \right] \)

We then showed that these expressions could be computed using Algorithms 2 and 3 and Eqn. C.10, C.11, C.12, and C.13.
C.2.3 M-Step

In the maximization step, we maximize the expected value of the complete data log likelihood with respect to the parameter vector \( \psi = (\gamma, \theta_0, \Sigma) \). In other words, we maximize \( Q(\psi | \psi^{(i)}) \) by solving for \( \nabla Q(\psi | \psi^{(i)}) = 0 \) in order to get updated estimates for \( \psi \), namely \( \hat{\psi}^{(i+1)} = (\hat{\gamma}^{(i+1)}, \hat{\theta}_0^{(i+1)}, \hat{\Sigma}^{(i+1)}) \).

**Solving for \( \hat{\gamma}^{(i+1)} \)**

We take the gradient of \( Q(\psi | \psi^{(i)}) \) with respect to \( \gamma \)

\[
\nabla Q(\psi | \psi^{(i)}) = \nabla \mathbb{E}[\log p(\theta, N | \psi^{(i)})] = \mathbb{E}[\nabla \log p(\theta, N | \psi^{(i)})] \tag{C.14}
\]

Thus we need to solve for \( \nabla \log p(\theta, N | \psi^{(i)}) \)

\[
\nabla \log p(\theta, N | \psi^{(i)}) = \nabla \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} (n_{k,l} \log(\lambda_k \Delta) - \lambda_k \Delta) - \frac{KR}{2} \log(2\pi) \\
- \frac{K}{2} \log |\Sigma| - \frac{1}{2} \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right] \\
= \nabla \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} (n_{k,l} \log(\lambda_k \Delta) - \lambda_k \Delta) \right] \\
= \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} (n_{k,l} \nabla(\log(\lambda_k \Delta)) - \nabla \lambda_k \Delta) \right]
\]

The partial derivatives of \( \log(\lambda_k \Delta) \) and \( \lambda_k \) with respect to \( \gamma_j \) are as follows

\[
\frac{\partial}{\partial \gamma_j'} (\log(\lambda_k \Delta)) = \frac{\partial}{\partial \gamma_j} \left( \log \Delta + \sum_{r=1}^{R} \theta_{k,r} g_r(l \Delta) + \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right) \\
= \frac{\partial}{\partial \gamma_j} \left( \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right) \\
= \sum_{j=1}^{J} \frac{\partial \gamma_j}{\partial \gamma_j'} n_{k,l-j} \\
= n_{k,l-j}
\]

and
\[
\frac{\partial}{\partial \gamma_j} \lambda_k = \frac{\partial}{\partial \gamma_j} \left( \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right) \\
= \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \cdot \frac{\partial}{\partial \gamma_j} \left( \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right) \\
= \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \cdot \frac{\partial}{\partial \gamma_j} \left( \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right) \cdot \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \\
= \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r(l\Delta) \right\} \cdot n_{k,l-j} \cdot \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \\
= n_{k,l-j} \lambda_k 
\]

Therefore, the gradient vector with respect to \( \gamma \) is

\[
\nabla \log p(\theta, N|\psi^{(i)}) = \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} n_{k,l-j'} - n_{k,l-j'} \lambda_k \right) \right]_{j'=1:J} \tag{C.15} 
\]

Thus, the gradient of \( Q(\psi|\psi^{(i)}) \) with respect to \( \gamma \) is

\[
\nabla Q(\psi|\psi^{(i)}) = \nabla \mathbb{E}[\log p(\theta, N|\psi^{(i)})] \\
= \mathbb{E}[\nabla \log p(\theta, N|\psi^{(i)}) || N, \psi^{(i)}] \\
= \mathbb{E} \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} n_{k,l-j'} - n_{k,l-j'} \lambda_k \right) \right]_{j'=1:J} \tag{C.15} \\
= \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} n_{k,l-j'} - n_{k,l-j'} \mathbb{E}[\lambda_k] \right)_{j'=1:J} 
\]

We thus need to solve for an expression for \( \mathbb{E}[\lambda_k || N, \psi^{(i)}] \) in order to find the gradient of \( Q(\psi|\psi^{(i)}) \). Using the fact that \( \theta_{k,r} \) is approximately normally distributed with mean \( \theta_{k|K,r} \) and variance \( W_{k|K,r,r} \), we have the following expression:
\[ \mathbb{E}[\lambda_k] = \mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r (l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right] \]
\[ = \mathbb{E} \left[ \exp \left\{ \sum_{r=1}^{R} \theta_{k,r} g_r (l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right] \]
\[ = \exp \left\{ \sum_{r=1}^{R} \left[ \theta_{k,K,r} + \frac{1}{2} W_{k,K,r,r} \right] g_r (l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \]

(C.16)

Thus, the gradient of \( Q(\psi|\psi^{(i)}) \) with respect to \( \gamma \) is
\[
\nabla Q(\psi|\psi^{(i)}) = \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} m_{k,l-j'} - n_{k,l-j'} \right) \exp \left\{ \sum_{r=1}^{R} \left[ \theta_{k,K,r} + \frac{1}{2} W_{k,K,r,r} \right] g_r (l\Delta) \right\} \right] \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \Delta \right)_{j'=1:J}
\]

(C.17)

Since this gradient doesn’t have a closed-form solution we use the Newton-Raphson algorithm to find updates to \( \gamma \) (see Algorithm 4). This algorithm also requires that we solve for \( \nabla^2 Q(\psi|\psi^{(i)}) \) with respect to \( \gamma \).

We therefore compute the Hessian matrix of \( Q(\psi|\psi^{(i)}) \) with respect to \( \gamma \). We denote \( \frac{\partial^2}{\partial \gamma_{j'} \partial \gamma_{j''}} Q(\psi|\psi^{(i)}) \) as \( Q(\psi|\psi^{(i)})_{j', j''} \), the \( j', j'' \) entry of the Hessian matrix. The following is a computation of each entry.

\[
\frac{\partial^2}{\partial \gamma_{j'} \partial \gamma_{j''}} Q(\psi|\psi^{(i)}) = \frac{\partial}{\partial \gamma_{j'}} \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} m_{k,l-j'} - n_{k,l-j'} \mathbb{E}[\lambda_k] \Delta \right) \right]
\]
\[ = \frac{\partial}{\partial \gamma_{j'}} \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( -n_{k,l-j'} \mathbb{E}[\lambda_k] \Delta \right) \right]
\]
\[ = \sum_{k=1}^{K} \sum_{l=1}^{L} \left( -n_{k,l-j'} \frac{\partial \mathbb{E}[\lambda_k]}{\partial \gamma_{j''}} \Delta \right)
\]
We therefore need to solve for \( \frac{\partial E[\lambda_k]}{\partial \gamma_{j''}} \) using the approximation from Eqn. C.16:

\[
\frac{\partial E[\lambda_k]}{\partial \gamma_{j''}} = \frac{\partial}{\partial \gamma_{j''}} \left( \exp \left\{ \sum_{r=1}^{R} \left[ \theta_{k|r,K,r} + \frac{1}{2} W_{k|K,r,r} \right] g_r(l\Delta) \right\} \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right)
\]

\[
= \exp \left\{ \sum_{r=1}^{R} \left[ \theta_{k|r,K,r} + \frac{1}{2} W_{k|K,r,r} \right] g_r(l\Delta) \right\} \frac{\partial}{\partial \gamma_{j''}} \left( \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\} \right)
\]

\[
= \exp \left\{ \sum_{r=1}^{R} \left[ \theta_{k|r,K,r} + \frac{1}{2} W_{k|K,r,r} \right] g_r(l\Delta) \right\} \left( \frac{\partial}{\partial \gamma_{j''}} \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right) \exp \left\{ \sum_{j=1}^{J} \gamma_j n_{k,l-j} \right\}
\]

\[
= n_{k,l-j'} E[\lambda_k]
\]

Thus, the \( j', j'' \) cell of the Hessian of \( Q(\psi|\psi^{(i)}) \) is:

\[
Q(\psi|\psi^{(i)})_{j',j''} = \sum_{k=1}^{K} \sum_{l=1}^{L} \left[ -n_{k,l-j'} \cdot n_{k,l-j''} E[\lambda_k] \Delta \right]
\]

And therefore, we can solve for the updates of \( \gamma \) using the Newton-Raphson Algorithm

---

**Algorithm 4 Newton-Raphson Algorithm**

1: \( \text{diff} = 100 \)
2: \( \gamma_{old} = \gamma^{(i)} \)
3: while \( \text{diff} > 10^{-2} \) do
4: \( \gamma_{new} = \gamma_{old} - [\nabla^2 Q(\psi|\psi^{(i)})]^{-1} \times [\nabla Q(\psi|\psi^{(i)})] \)
5: \( \text{diff} = \max(|\gamma_{new} - \gamma_{old}|) \)
6: \( \gamma_{old} = \gamma_{new} \)
7: end while
8: \( \gamma^{(i+1)} = \gamma_{new} \)
Solving for $\hat{\Sigma}^{(i+1)}$

We start solving for $\hat{\Sigma}^{(i+1)}$ by finding the gradient of $Q(\psi|\psi^{(i)})$ with respect to $\Sigma$ requires solving for $\nabla \log p(\theta, N|\psi^{(i)})$ with respect to $\Sigma$ (see Eqn. [C.14]):

$$
\nabla \log p(\theta, N|\psi^{(i)}) = \nabla \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} \log(\lambda_k \Delta) - \lambda_k \Delta \right) - \frac{KR}{2} \log(2\pi) 
- \frac{K}{2} \log |\Sigma| - \frac{1}{2} \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right] 
$$

$$
= -\frac{K}{2} \nabla \left[ \log |\Sigma| \right] - \frac{1}{2} \nabla \left[ \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right] 
$$

We therefore need to solve for the two partial derivatives. The derivation of the partial derivative of the first term is as follows:

$$
\frac{\partial}{\partial \Sigma_{a,b}} \log |\Sigma| = \frac{1}{|\Sigma|} \left( \text{adj}(\Sigma) \right)_{b,a} 
$$

since $\frac{\partial \log(f(x))}{\partial(x)} = \frac{1}{f(x)} \frac{\partial f(x)}{\partial x}$

by Jacobi’s Formula

$$
= (\Sigma^{-1})_{b,a} 
$$

since $A^{-1} = \frac{1}{|A|} \text{adj}(A)$

$$
= (\Sigma^{-1})_{a,b} 
$$

since $\Sigma^T = \Sigma$

We then solve for the second term noting that $\Sigma^T$ is symmetric:

$$
\frac{\partial}{\partial \Sigma_{a,b}} \left[ \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right] 
= \sum_{k=1}^{K} \frac{\partial}{\partial \Sigma_{a,b}} \left[ (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \right] 
$$

$$
= \sum_{k=1}^{K} -\Sigma^{-T} (\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \Sigma^{-T} 
$$

$$
= \sum_{k=1}^{K} -\Sigma^{-1} (\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \Sigma^{-1} 
$$

We then use these two terms to solve for $\mathbb{E}[\nabla \log p(\theta, N|\psi^{(i)})] = 0$:  


\[
0 = \mathbb{E} \left[ -\frac{K}{2} \Sigma^{-1} + \frac{1}{2} \sum_{k=1}^{K} \Sigma^{-1}(\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \Sigma^{-1} \right]
\]

\[
0 = -\frac{K}{2} \mathbb{E}[\Sigma^{-1}] + \frac{1}{2} \sum_{k=1}^{K} \mathbb{E} \left[ \Sigma^{-1}(\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \Sigma^{-1} \right] \quad \mathbb{E}[a + b] = \mathbb{E}[a] + \mathbb{E}[b]
\]

\[
\frac{K}{2} \Sigma^{-1} = \frac{1}{2} \sum_{k=1}^{K} \mathbb{E} \left[ \Sigma^{-1}(\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \Sigma^{-1} \right]
\]

\[
K = \sum_{k=1}^{K} \mathbb{E} \left[ (\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \Sigma^{-1} \right] \quad \text{left multiply by } 2\Sigma
\]

\[
K \Sigma = \sum_{k=1}^{K} \mathbb{E} \left[ (\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \right] \quad \text{right multiply by } \Sigma
\]

\[
\Sigma = \frac{1}{K} \sum_{k=1}^{K} \mathbb{E} \left[ (\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \right] \quad \text{right multiply by } \Sigma
\]

Therefore, we have a closed form update function for \(\Sigma^{(i+1)}\), namely:

\[
\Sigma^{(i+1)} = \frac{1}{K} \sum_{k=1}^{K} \mathbb{E} \left[ (\theta_k - \theta_{k-1})(\theta_k - \theta_{k-1})^T \right]
\] (C.18)

**Solving for \(\hat{\theta}_0^{(i+1)}\)**

Again, we start solving for \(\hat{\theta}_0^{(i+1)}\) by finding the gradient of \(Q(\psi|\psi^{(i)})\) with respect to \(\theta_0\) and see that it requires solving for \(\nabla \log p(\theta, N|\psi^{(i)})\) with respect to \(\theta_0\) noting that \(\lambda_k\) is independent of \(\theta_0\) (see Eqn. C.14):

\[
\nabla \log p(\theta, N|\psi^{(i)}) = \nabla \left[ \sum_{k=1}^{K} \sum_{l=1}^{L} \left( n_{k,l} \log(\lambda_k \Delta) - \lambda_k \Delta \right) - \frac{KR}{2} \log(2\pi) \right. \\
- \frac{K}{2} \log |\Sigma| - \frac{1}{2} \sum_{k=1}^{K} (\theta_k - \theta_{k-1})^T \Sigma^{-1} (\theta_k - \theta_{k-1}) \left. \right]
\]

\[
= -\frac{1}{2} \nabla \left[ (\theta_1 - \theta_0)^T \Sigma^{-1} (\theta_1 - \theta_0) \right]
\]

\[
= \Sigma^{-1} (\theta_1 - \theta_0)
\]

The last step was simplified since \(\Sigma^{-1}\) is symmetric.
Therefore, we can solve $\nabla Q(\psi \mid \psi^{(i+1)}) = 0$ to find $\theta_0^{(i+1)}$:

$$\nabla Q(\psi \mid \psi^{(i+1)}) = 0$$

$$\mathbb{E}[\Sigma^{-1}(\theta_1 - \theta_0)] = 0$$

$$\mathbb{E}[\Sigma^{-1}\theta_1] - \mathbb{E}[\Sigma^{-1}\theta_0] = 0$$

$$\mathbb{E}[\Sigma^{-1}\theta_0] = \mathbb{E}[\Sigma^{-1}\theta_1]$$

$$\Sigma^{-1}\theta_0 = \Sigma^{-1}\mathbb{E}[\theta_1]$$

$$\theta_0 = \mathbb{E}[\theta_1 \mid N, \psi^{(i)}]$$

left multiply by $\Sigma$

Thus, thus we have a closed form update equation for $\theta_0$, namely

$$\theta_0^{(i+1)} = \mathbb{E}[\theta_1 \mid N, \psi^{(i)}] = \theta_1|_{K}$$  \hspace{1cm} (C.19)

Summary

We have derived the update equations for each of the unknown parameters in $\psi$. In particular, we use the Newton-Raphson algorithm to solve for $\gamma^{(i+1)}$ (see Algorithm 4) and a closed form equation for $\Sigma^{(i+1)}$ and $\theta_0^{(i+1)}$ (see Eqn. C.18 and C.19, respectively).

C.2.4 EM Algorithm Summary

In the previous sections, we derived the equations necessary to run the EM algorithm. We now summarize all of the equations into a single algorithm (Alg. 5) that is used to find the unobservable parameters $\theta_k$ and the unknown parameters $\psi = (\gamma, \theta_0, \Sigma)$.

Algorithm 5 EM Algorithm for SS-GLM

Initialize $\psi^{(1)}$

Set $i = 1$, $diff = 100$, $Q_{old} = 100$

while $diff > 10^{-2}$ do

Run E-Step I: Linear Filtering Algorithm (Alg. 2) using $\psi^{(i)}$

Run E-Step II: Fixed-Interval Smoothing Algorithm (Alg. 3) using $\psi^{(i)}$

Run E-Step III using $\psi^{(i)}$

Compute $Q(\psi \mid \psi^{(i)})$

Compute $\gamma^{(i+1)}$ with Newton-Raphson Algorithm (Alg. 4)

Compute $\Sigma^{(i+1)}$ with Eqn. C.18

Compute $\theta_0^{(i+1)}$ with Eqn. C.19

Compute $diff = Q(\psi \mid \psi^{(i)}) - Q_{old}$

Set $i = i + 1$, $Q_{old} = Q(\psi \mid \psi^{(i)})$

end while
Bibliography


